REPORT DOCUMENTATION PAGE

AFRL-SR-AR-TR-04-

0544

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From – To)*
30 September 2004	Final Report	From 1 July 2001 to 30 Sep 2004

4. TITLE AND SUBTITLE
Robust Adaptive Control of UCAVs

5a. CONTRACT NUMBER
F49620-01-C-0027

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Yutaka Ikeda, James Ramsey, Eugene. Lavretsky, Patrick McCormick

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
McConnell Douglas Corporation
Wholly-Owned Subsidiary of The Boeing Company
P.O. Box 516
St. Louis, MO 63166-0516

8. PERFORMING ORGANIZATION REPORT NUMBER
2004P0037

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Air Force Office of Scientific Research
Directoate of Dynamics and Control
AFOSR/~~NA~~ NM
801 N. Randolph Street
Arlington, VA 22203

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

20041105 149

14. ABSTRACT
This report describes work performed to investigate adaptive nonlinear control for UCAVs. Specifc control techniques investigated were Internal Model Principle Nonliear Output Regulation for UCAV flight control desing and aero-servo-elastic filter deisgn; and Adaptive Inverse Control for electromagnetic actuator control design for UAVs and advaced missiles. Simulation results show good command following (or reference tracking) characteristic of the Internal Model Priciple Output Regulation technique for designs of flight control and actuator control. The simulation results also show good disturbance attenuation characteristic of the Internal Model Principle Output Regulation technique for aero-serve-elastic filter design. Neural Networks Adaptive Control was also evaluated in the UCAV flight control application, and similar results to those with the Internal Model Priciple Output Regulation control were observed. The Adaptive Inverrse Control technique showed some potential for comtrolling the electromagnetic actuators which contain nonliear elemets such as dead-zone, backlash, friction, and quantization errors. However, it introduced chattering and sensitivity to the initial estimation error of the unknow parameters. The Internal Model Priciple Output Regualtion technique was also applied to the eletromagnetic actuator model and resolved the chattering and sensitivity to the parameter estimation errors.

15. SUBJECT TERMS
Internal Model Principle, Nonlinear Output Regulation, Adaptive Control, Neural Networks, Adaptive Inverse Control

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Yutaka Ikeda
Unclassified	Unclassified	Unclassified	UL	98	19b. TELEPHONE NUMBER *(Include area code)* (314) 232-3974

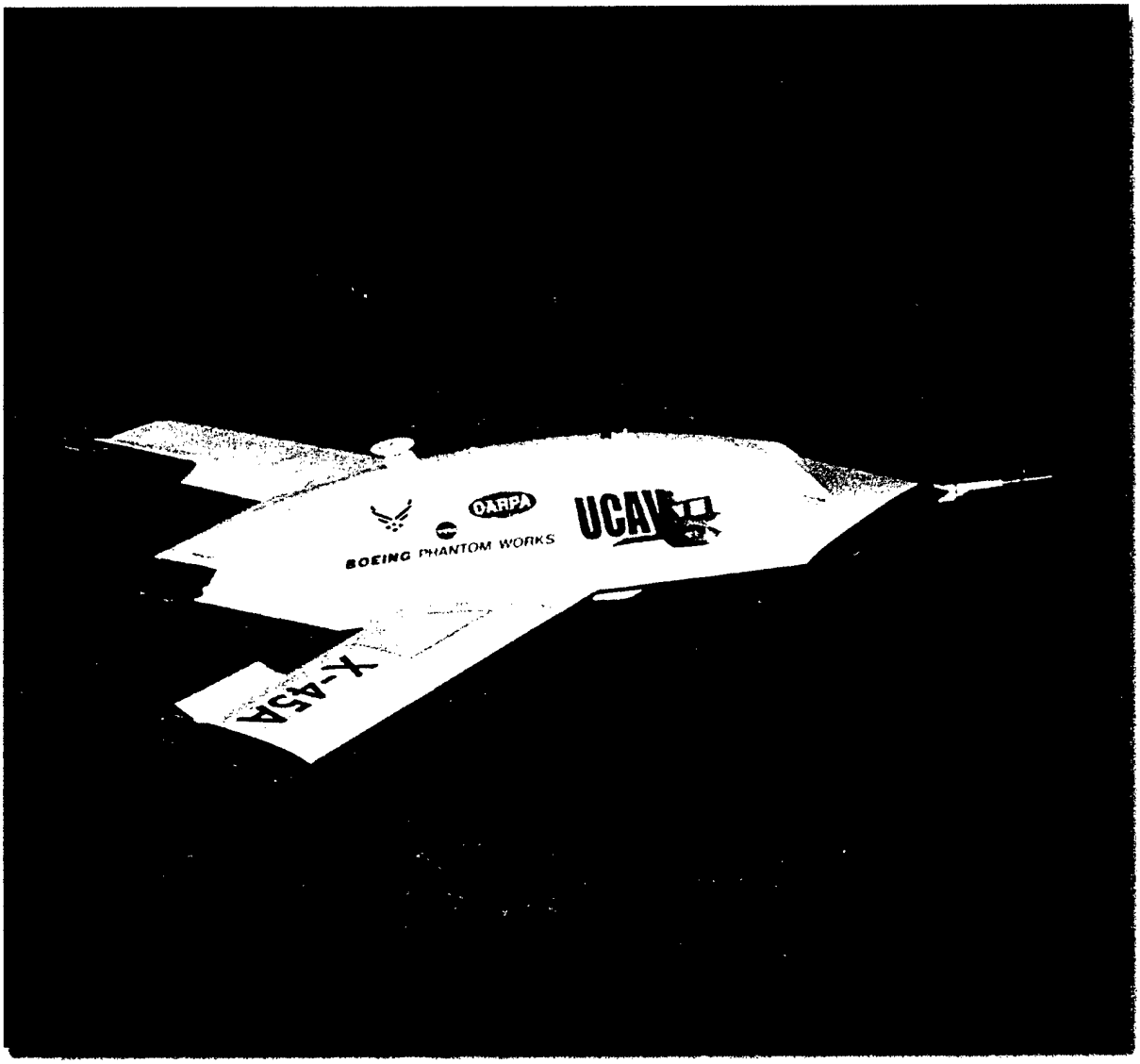
Standard Form 298 (Rev 8/98)
Prescribed by ANSI Std. Z39.18

BOEING-STL 2004P0037
30 September 2004

ROBUST ADAPTIVE CONTROL OF UCAVS

FINAL REPORT

Contract Number F49620-01-C-0027



BOEING

TABLE OF CONTENTS

Page

LIST OF FIGURES

	Title	Page

1.0 EXECUTIVE SUMMARY

This report documents the results from the research performed under the contract F49629-01-C-0027 entitled "Robust Adaptive Control of UCAVs". Our research was aimed at developing robust nonlinear adaptive control algorithms for UCAV flight control, flexible mode compensation and stabilization, and electromagnetic actuators. Specific techniques investigated were Internal-Model-Principle-Based Nonlinear Output Regulation and the Adaptive Inverse Method. For comparison purposes, Neural Networks was also investigated.

1.1 Robust Adaptive Nonlinear Output Regulation

In order to develop nonlinear adaptive control for UCAVs, Internal Model Principle Nonlinear Output Regulation (IMPNOR) technique was investigated. This controller design technique was applied to structural mode filter design, and UCAV adaptive flight control, and the performance was evaluated for implementation.

For the structural mode filter design, the classical notch filter is often used in practice. The draw back to this approach is that the frequency to be suppressed needs to be known exactly, otherwise the filter would miss the target frequency and fail to suppress it. In addition, once the filter has been designed, the target frequency would be fixed and cannot accommodate any other frequencies. As will be discussed in the next section, the frequency that needs to be attenuated could change depending on maneuvers the aircraft performs, particularly for flexible aircraft such as UCAVs. Hence, the goal was to develop an adaptive method to dynamically filter out the undesirable effects of the structural modes on the flight control sensors. As will be discussed in Section 3, the IMPNOR method was employed to design a control law that tracks the reference command for take-off and landing while suppressing the undesirable harmonic signals whose frequencies were not known *a priori.* The simulation results show that simultaneous command tracking and disturbance rejection were achieved successfully by the IMPNOR method.

For the UCAV adaptive flight control, we applied the IMPNOR method to the X-45A linear model and designed flight control laws which accommodate surface failures and nonlinear modeling errors. Both inner-loop and outer-loop responses were evaluated with and without surface failures. The IMPNOR method tracked the reference command successfully in spite of the surface failures. To assess the performance of the resulting IMPNOR control laws, an alternative method using Neural Networks (NN) was applied to the same linear model and the results were compared. Both IMPNOR and NN controllers seemed to track the command well. The details are discussed in Section 4.

1.2 Adaptive Control of Electromagnetic Actuators

The Adaptive Inverse Control method developed by G. Tao and P. Kokotovic was investigated to design a controller for an electromagnetic actuator (EMA). Specifically, we applied the method to compensate the nonlinear dynamics that appear in the Miniaturized Munition Technology (MMT) actuator subsystems known as dead-zone, Coulomb friction, and quantization errors. The simulation results showed the Adaptation Inverse Control method performed better than a detuned fixed inverse controller or a controller with no inverse model associated with it. However, the Adaptive Inverse Control method also introduced chattering, slow adaptation process, and less robustness to larger magnitude of initial parameter estimation errors. To alleviate these problems we considered the IMPNOR technique as an alternative

method, and applied it to MMT actuator subsystems. The simulation results showed that IMPNOR resolved the chattering and robustness issues while tracking the command in the presence of harmonic disturbances. The details are discussed in Section 5.

1.3 Aircraft Departure Analysis/Prediction Tool

We investigated a robustness analysis method appeared in [10] in an attempt to develop an Aircraft Departure Analysis/Prediction Tool. We also visited Dr. R. Murray and Dr. J. Doyle at California Institute of Technology, who are two of the co-authors of [10] to discuss the method. After the discussion, and further investigation, we decided that the method is not mature enough at the present time for technology transition, and that we will not proceed any further on this task.

1.4 Research Collaboration with Universities

Throughout the entire contract period of performance for this research program, Boeing and Washington University (Dr. C. Byrnes and Dr. A. Ishidori) had a close working relationship and shared the interest in applying the Internal Model Principle Nonlinear Output Regulation Control technique to real aircraft system in an effort to transition the technology emanating from this research program. The research collaboration between Boeing and Washington University culminated in publication of [11].

We also had a dialogue for potential research collaboration with Dr. Murray at California Institute of Technology (CalTech) in the area of robustness analysis techniques. Boeing and CalTech had a technical coordination meeting where both parties expressed interest in collaboration, and decided that we keep searching for research common ground by keeping an open dialogue.

Boeing also had research collaboration with Dr. Hovakimyan at Virginia Polytechnic Institute and State University in the area of Air Collision Avoidance problem from the Control Theory point of view. This research collaboration was culminated in publication of [12].

2.0 INTRODUCTION

Affordability and safety have become the prime focus for today's defense research. New Unmanned Combat Air Vehicles (UCAVs) are being developed to significantly reduce the cost compared to existing manned systems. Smart GPS guided weapons are being developed to improve mission effectiveness and reduce costs per target kill. To answer to these challenging demands and/or needs, our research was centered around UCAVs and advanced munitions such as Joint Direct Attack Munitions (JDAM) and the Small Smart Bomb (250 lb weapon under development), and the efforts were focused on the development of new control algorithms that would improve flight safety, mission reliability, and cost reduction for UCAVs and these advanced weapon systems.

The specific areas of primary investigation under this AFOSR program were:

- Robust adaptive control algorithms for flight control
  - Robust adaptive control laws
  - Adaptive aeroservoelastic compensation for flexible aircraft
- Adaptive control compensation for electric actuators
- Aircraft departure analysis and prediction tools

These investigations were performed using new theory on robust output regulation for nonlinear systems being developed at Washington University in St. Louis (Byrnes and Isidori), robust adaptive control methods developed at UC Santa Barbara (Kokotovic), and robustness analysis of nonlinear systems developed at California Institute of Technology (Murray).

Robust adaptive output regulation [1]-[5] offers a potential to enhance flight performance and improve robustness to uncertain aerodynamics and structural dynamics. Our research investigated this new approach [5], and compared it with the adaptive neural networks approach using an X-45A linearized model. One of our goals was to demonstrate this technology in a flight control application, and to extend it to design an adaptive aeroservoelastic compensator for flexible aircraft, which would fulfill a critical need currently not present in industry.

Our research on adaptive control compensation for electric actuators is directly applicable to both UCAV and JDAM. UCAV is using electric actuators that are a derivative of the JDAM design. Our research in this area would significantly reduce the cost of these actuators. Each UCAV uses 18 actuators, and each JDAM uses 3 actuators. Our success would lead to reduced parts count, reduced weight, and reduced cost for each actuator. Multiplying by the production numbers for JDAM, and the potential savings from UCAV and the other munitions programs that are under development, the potential savings are enormous.

In addition to being affordable, UCAV must be reliable and safe, and must have the ability to integrate with manned aircraft operations. The missions to be performed by UCAV are highlighted in Figure 2-1.

Robust adaptive control and self-diagnostics, prognostics, and health monitoring technologies are needed to give UCAV the same reliability and safety of manned aircraft.

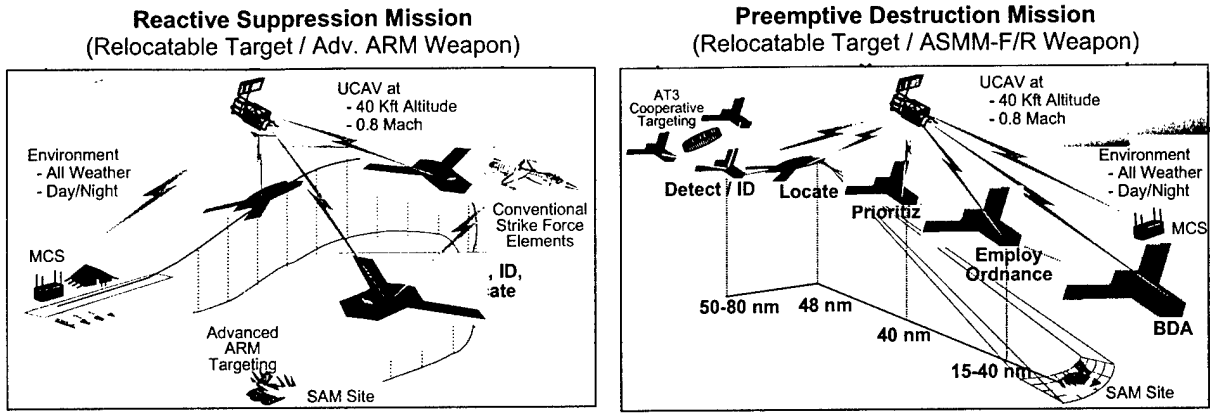
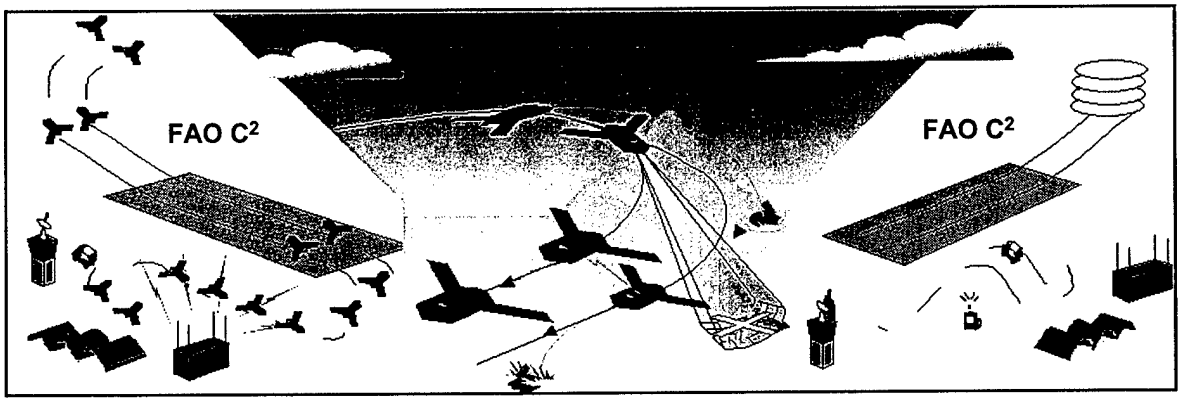


Figure 2-1. UCAV Suppression of Air Enemy Defense Mission Scenarios

Autonomous ground operations, robust flight control, and multi-vehicle coordinated operations are being designed for UCAV to minimize operator workload, increase mission reliability, and reduce support costs. These new levels of vehicle autonomy introduce new requirements on the aircraft's control system. Sophisticated contingency management algorithms must replace the pilot's ability to adapt to unforeseen and unpredictable events to guarantee flight safety and mission reliability. The aircraft's control system must be able to respond to these contingencies. Figure 2-2 illustrates the guidance and control modes for UCAV, as well as several of the contingency modes. The figure illustrates autonomous ground operations (auto taxi and takeoff), ingress to target area, mission execution, and safe return to base.



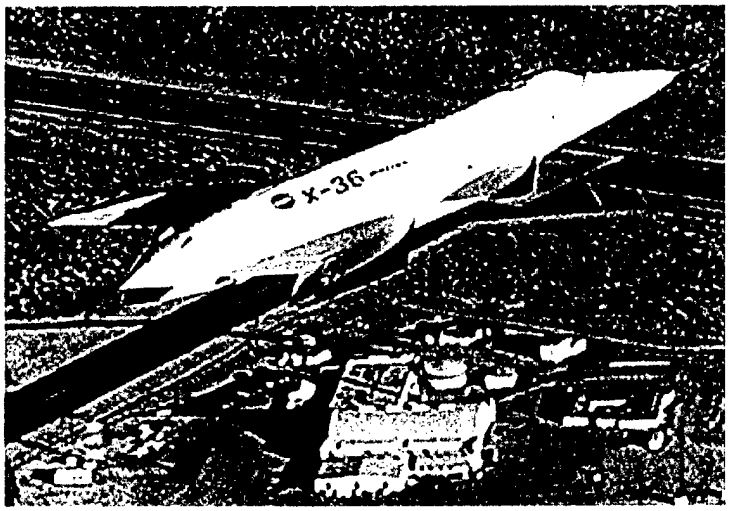
	Standby	Takeoff	Flight	Approach
Flight modes	• Off • Engine on • Maintenance • Pre/Post • Mission test	• Taxi • Pre-Takeoff test • Hold • Takeoff roll • Initial climb	• Climb • Cruise • Descent	• Go around • Holding pattern • Runaway pattern • Flare • Rollout • Taxi
Contingency modes		• About initial climb • Refuse takeoff • Others	• Return to base • Emergency return • Others	• Emergency return • Emergency stop • Others

GP44275002.ppt

Figure 2-2. UCAV Guidance, Control, and Contingency Modes

Robust adaptive control plays a critical role in the autonomous control of UCAVs. These algorithms must accommodate unknown or uncertain aerodynamics, flight control actuator failures and/or battle damage. Our research addresses the need for flight control and for adaptive aeroservoelastic compensation. The need for reconfigurable flight control is well understood, and alternate solutions to this problem exist. The need for adaptive aeroservoelastic compensation for flexible aircraft remains unanswered.

The X-36 Agility Research Aircraft shown in Figure 2-3 is a 28% scale test aircraft used to demonstrate agility and handling qualities in a tailless fighter (unstable in multiple axes). The X-36 is all composite aircraft that proved to have structural modes that depend upon flight condition. Due to both the aerodynamic loading on the fuselage and wing and the control surface deflection, the structural modes varied in frequency and in damping. Designing compensation to attenuate the effects of the structural modes on the flight control sensors (gyros and accelerometers) proved to be costly in time and manpower, and introduced risk into the program. The resulting compensation used filters that were scheduled with flight condition, including control surface deflection.



GP44275003.ppt

Figure 2-3. X-36 Agility Research Aircraft

Key in making the Boeing UCAV design affordable has been the incorporation of all-electric aircraft technologies. Existing aircraft use hydraulic actuation systems that are expensive, heavy, and require significant maintenance. The Boeing UCAV design currently has 18 electro-magnetic actuators (EMAs). Six electric actuators are used to control UCAV's flight control surfaces, as shown in Figure 2-4, with 2 actuators used to control the thrust vectoring system, and additional 10 actuators to control the nose and main landing gear, gear doors, steering system, and weapon bay doors.

UCAV has baselined identical actuators for all 18 EMAs. This in itself (using the same actuator everywhere on the aircraft) greatly reduces aircraft costs and logistics footprint. These UCAV electric actuators are manufactured by HR Textron under subcontract to Boeing. The HR Textron UCAV design is a derivative of the Joint Direct Attack Munition (JDAM) actuator design, and uses common parts and manufacturing processes and assembly. The large production rate from the JDAM program makes the HR Textron UCAV actuator very attractive from an affordability standpoint.

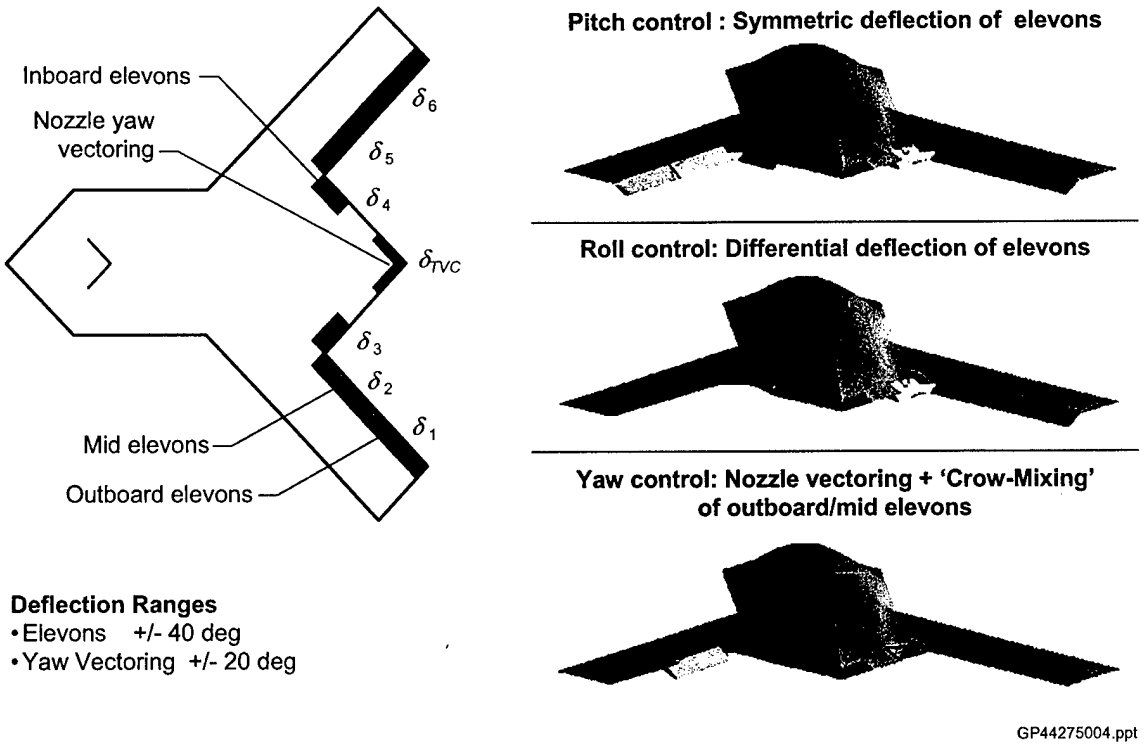


Figure 2-4. UCAV Flight Control Surfaces and Actuators

The UCAV actuator design, shown in Figure 2-5, is an electromechanical design, in which two brushless DC motors on a common shaft drive a ball screw to convert rotary motion into linear motion. When assembling the ball screw in the actuator unit, small ball bearings are placed in between the ball screw and the ball nut. After the actuator is assembled the backlash is measured. If the system is too sloppy, the unit is disassembled and slightly larger balls are inserted to remove the slop. This is labor intensive and expensive. We would like to assemble the unit once and for all, and accept whatever the tolerances are there. Allowing larger tolerances in the machining of the ball screw could further reduce actuator costs. Relaxing these tolerances will lead to a system that has more backlash, however.

Thermal expansion effects also complicate the design of the actuator. If lower cost materials could be used (that have greater thermal expansion), then costs could be further reduced. These materials are also easier to machine, reducing tooling costs.

The two-DC-motor design shown in Figure 2-5 exhibits the following characteristics:

- Stall load: 2,500 lb
- Max load prior to back drive: 3,500 lb
- Max brake hold capability: 5,000 lb (unpowered)
- 4" and 6" total stroke configurations
- 6 in/sec no load rate, 4.5 in/sec at 1,500 lb
- 250 flight hour actuator life

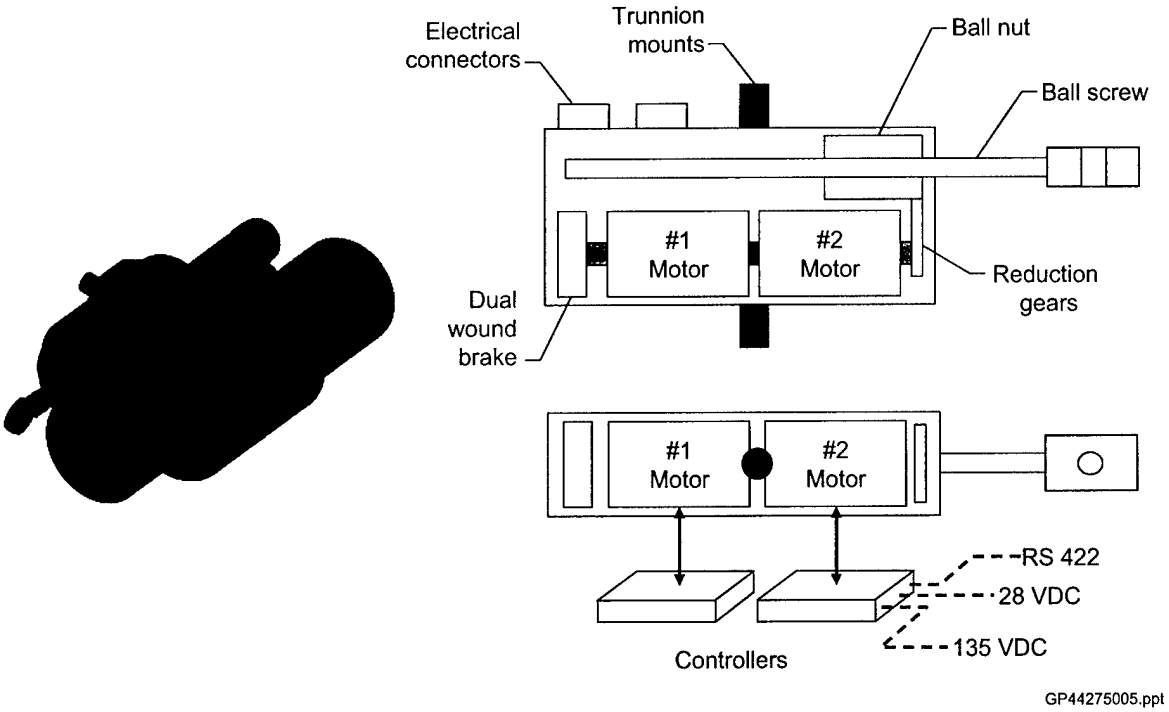


Figure 2-5. UCAV Electric Actuator

- 28 VDC and 135 VDC
- Peak current: 12 amps per motor (24 amps per actuator)
- Estimated weight: 10 lb
- One Brake, dual coil (Reliability if one fails)
- Two DC motors torque summed on common shaft
- Gear reduction to ball screw providing linear output

This type of EMA actuator exhibits different types of nonlinearity such as dead-zone, backlash, and Coulomb/Viscous friction. Our research focused on adaptive compensation to remove/negate these nonlinear effects, and produced an actuator controller that has repeatable performance from unit to unit.

Finally, this research program under AFOSR contract addresses fundamental scientific issues with a goal of rapid transition to real applications. The combination of "technology push" by basic research and "technology pull" by the applied programs provides specific technology transition objectives and paths. This work is directly linked to programs at The Boeing Company and US Air Force.

3.0 ROBUST ADAPTIVE NONLINEAR OUTPUT REGULATION
- An Application To Structural Mode Filter -

In this section, we discuss the Internal Model Principle Nonlinear Output Regulation (IMPNOR) method that suppresses the effect of harmonic disturbances on the regulated output of a system to be controlled. Controllers which are optimal in an LQ senses are derived for the suppression of harmonic disturbances for discrete-time, finite-dimensional, multivariable linear systems. These controllers are shown to enjoy some of the robustness features concomitant with an LQ optimization scheme.

This work can be applied in a variety of settings, for example in active noise control which was studied in [9] for scalar-input scalar-output systems using a minimum variance performance measure. The suppression of harmonic disturbances also arises in the active control of vibrations in helicopters[6], where a continuous-time model is used in the context of LQ control and Kalman filtering.

Our interest in suppression of harmonic disturbances arose in the development of feedback control strategies for next generation aircraft in particular, for certain aircraft it is known that the wings exhibit slightly damped flexible behavior which, when excited by a wind gust, produce a disturbance which additively corrupts some important velocity measurements. The frequencies of the slightly damped (exogeneous) signal generator are known, but the wind gust may be modeled as a dirac delta function which forces the exogeneous system into an unknown initial state. The control objective is to track a prescribed trajectory while suppressing the disturbance produced by the exogenous system. This is a slight modification of the standard problem of output regulation, in which the reference trajectory itself is also assumed to be generated by an exosystem. As we shall show, this modification is easily accommodated within the present framework of output regulation. However, since output regulation ensures some asymptotically stability of the error while the tracking and disturbance rejection problem requires a finite time horizon in practice, we have assumed the exosystem is undamped and consists of a finite bank of harmonic oscillators.

As part of an ongoing research effort, we are developing a solution to the problem for a nonlinear system which incorporates both the rigid body dynamics and aeroelastic dynamics. In this section, we illustrate our use of the internal model principle[8] to solve this problem for continuous time linear systems. Specifically, in Section 3.2 we introduce the basics of internal model based design for a scalar-input, scalar-output system. In Section 3.3, we illustrate this design for two particular systems where the desired trajectory idealizes a take-off and landing maneuver. In Section 3.4 we give some brief remarks about the internal model principle. One remarkable feature of this principle is its ability to systematically produce classical design approaches in a variety of particular control problems. In the case at hand, it produces a notch filter for the transfer function from the disturbance to the tracking error, with notches, or (blocking) zeroes, at the natural frequencies of the exogenous system. Of course, the real power of this method is its ability to produce nonclassical, multivariable designs. Indeed, in Section 3.4 we consider three-input, three-output, minimum phase systems with vector relative degree (2; 2; 2). Interestingly, the internal model based controller design leads to a Linear Matrix Inequlaity (LMI) constraint on the design parameters, yielding a convex problem which is easily solved.

3.1 The Case of a SISO System

Consider a linear single-input single-output system having relative degree two and asymptotically stable zero dynamics. Any system of this kind can always be put, after a suitable change of coordinates in the state space, in the form

$$\begin{aligned} \dot{z} &= Az + B_1 x_1 + B_2 x_2 \\ \dot{x}_1 &= x_2 \\ \dot{x}_2 &= Cz + D_1 x_1 + D_2 x_2 + Ku \\ y &= x_1 \end{aligned} \tag{3.1}$$

in which $K \neq 0$ and the matrix A is Hurwitz.

It is well-known that single-input single-output systems having relative degree two and stable zero dynamics can always be asymptotically stabilized by means of a proportional-derivative output feedback, i.e. by means of a control law of the form

$$u = k_1 y + k_2 \dot{y} .$$

It is also well-known that a proportional-derivative output feedback can be used to solve the problem of having the output y(t) to track a prescribed trajectory $y_{ref}(t)$, if the latter and its first and second derivatives are available in real-time.

In this section, we describe how the problem of tracking a prescribed trajectory $y_{\text{ref}}(t)$ can be solved, for a system of the form (3.1), in case the information about the output y (i.e. x_1) is noise-free, but the information about the *rate of change y* (i.e. x_2) is corrupted by *harmonic noise*, i.e. if only x_1 and the quantity

$$x_{2,\text{noisy}} = x_2 + d , \tag{3.2}$$

where d is a superposition of sinuosoidal functions of time that are available for feedback.

Define

$$\begin{aligned} e_1 &= y - y_{\text{ref}} = x_1 - y_{\text{ref}} \\ e_2 &= \dot{y} - \dot{y}_{\text{ref}} = x_2 - \dot{y}_{\text{ref}} \end{aligned}$$

in which e_1 is indeed the tracking error and e_2, its rate of change. Then, system (3.1) can be rewritten as

$$\begin{aligned} \dot{z} &= Az + B_1(e_1 + y_{\text{ref}}) + B_2(e_2 + \dot{y}_{\text{ref}}) \\ \dot{e}_1 &= e_2 \\ \dot{e}_2 &= Cz + D_1(e_1 + y_{\text{ref}}) + D_2(e_2 + \dot{y}_{\text{ref}}) + Ku - \ddot{y}_{\text{ref}} . \end{aligned} \tag{3.3}$$

Let z_{rcf} be any solution of

$$\dot{z}_{\text{ref}} = Az_{\text{ref}} + B_1 y_{\text{ref}} + B_2 \dot{y}_{\text{ref}} \tag{3.4}$$

and define

$$e_z = z - z_{\text{ref}} .$$

Then system (3.3) can be rewritten as

$$\begin{aligned} \dot{e}_z &= Ae_z + B_1 e_1 + B_2 e_2 \\ \dot{e}_1 &= e_2 \\ \dot{e}_2 &= Ce_z + Cz_{\text{ref}} + D_1(e_1 + y_{\text{ref}}) + D_2(e_2 + \dot{y}_{\text{ref}}) + Ku - \ddot{y}_{\text{ref}} . \end{aligned} \tag{3.5}$$

Choose now a control of the form

$$\begin{aligned} u &= u(x_1, x_{2,\text{noisy}}, y_{\text{ref}}, \dot{y}_{\text{ref}}, \ddot{y}_{\text{ref}}, z_{\text{ref}}) \\ &= \tfrac{1}{K}[-Cz_{\text{ref}} - (D_1 + k_1)y_{\text{ref}} - D_2\dot{y}_{\text{ref}} + \ddot{y}_{\text{ref}} - D_1(x_1 - y_{\text{ref}}) - k_2(x_{2,\text{noisy}} - \dot{y}_{\text{ref}}) + v] \end{aligned}$$

in which v is an additional input to be determined later. With this control u, system (3.5) becomes

$$\begin{aligned} \dot{e}_z &= Ae_z + B_1 e_1 + B_2 e_2 \\ \dot{e}_1 &= e_2 \\ \dot{e}_2 &= Ce_z - k_1 e_1 - \bar{k}_2 e_2 - k_2 d + v , \end{aligned} \tag{3.6}$$

where

$$\bar{k}_2 = k_2 - D_2 .$$

It is known that *d(t)*, which is superposition of sinusoidal functions, can be put in the form

$$d(t) = \Psi^T w(t)$$

with *w(t)* solution of the homogeneous equation

$$\dot{w}(t) = Sw(t) \tag{3.7}$$

in which S is a matrix of the form

$$S = \begin{pmatrix} S_1 & 0 & \cdot & 0 \\ 0 & S_2 & \cdot & 0 \\ \cdot & \cdot & \cdots & \cdot \\ 0 & 0 & \cdot & S_r \end{pmatrix}$$

and

$$S_i = \begin{pmatrix} 0 & \omega_i \\ -\omega_i & 0 \end{pmatrix}$$

Under our assumption that the zero dynamics is either vacuous or is asymptotically stable, we know that any transmission zeros of the plant to be controlled lie in the left half plane. Since the eigenvalues of the exosystem (3.7) that generate the disturbance lie on the imaginary axis, we see that the standard nonresonance conditions for solution of the regular problem are satisfied.

Matters being so, the output regulation problem for $y_{ref} = 0$ is solvable, when the additional input v is chosen as the output of an *internal model* of the form

$$\begin{aligned} \dot{\xi} &= S\xi - Gk_1 e \\ v &= \Psi^T \xi \end{aligned}$$

where G is a vector of free design parameters. We claim that this also yields a solution for nontrivial y_{ref}. Indeed, introduction of v yields a closed loop system which, changing the variable ξ into

$$\chi = \xi - k_2 w$$

is described by set of equations

$$\begin{aligned} \dot{e}_z &= Ae_z + B_1 e_1 + B_2 e_2 \\ \dot{\chi} &= S\chi - Gk_1 e_1 \\ \dot{e}_1 &= e_2 \\ \dot{e}_2 &= Ce_z + \Psi^T \chi - k_1 e_1 - \bar{k}_2 e_2 \,. \end{aligned} \tag{3.8}$$

At this point, it remains to show that, if k_1, k_2 (or, what is the same, $\bar{k}_2$), and G is appropriately chosen, system (3.8) is asymptotically stable. If this is the case, in fact, we have in particular

$$\lim_{t\to\infty} e_1(t) = 0 \,,$$

which is the required tracking goal, regardless of the fact that the measurement of y is corrupted by noise.

To provide our stability result, set

$$\Psi^T = \begin{pmatrix} \Psi_1^T & \Psi_2^T & \cdots & \Psi_r^T \end{pmatrix}$$

and

$$G^T = \begin{pmatrix} G_1^T & G_2^T & \cdots & G_r^T \end{pmatrix}$$

where the partitions indicated are consistent with those of S.

Then, the following result holds.

Proposition 3.1: *Consider system (3.8). Choose $G_1, \ldots, G_r$ in such a way that*

$$\Psi_i^T S_i^{-1} G_i > 0 \,, \qquad i = 1, \ldots, r \tag{3.9}$$

$$\sum_{i=1}^{r} \Psi_i^T S_i^{-1} G_i < 1 \tag{3.10}$$

which is always possible. If $k_1 > 0$, $\bar{k}_2 > 0$ are sufficiently large, the system is asymptotically stable.

The proof of this Proposition is a straightforward consequence of the following Lemma, proven for instance in [7].

Lemma 3.2: *Consider the matrix*

$$J = \begin{pmatrix} A & 0 & \cdot & 0 & 0 & B \\ 0 & S_1 & \cdot & 0 & 0 & P_1 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & 0 & \cdot & S_r & 0 & P_r \\ 0 & 0 & \cdot & 0 & 0 & P_0 \\ C & Q_1 & \cdot & Q_r & Q_0 & -k \end{pmatrix} \tag{3.11}$$

If

$$P_i Q_i < 0 \qquad for\ all\ i = 0, \ldots, r \tag{3.12}$$

there exists k such that, if k > k*, the matrix J is Hurwitz.*

To provide the Proposition, we change coordinates in system (3.8) using

$$\begin{aligned} \tilde{e}_x &= e_x + A^{-1}B_1 e_1 \\ \tilde{\chi} &= \chi - k_1 S^{-1} G e_1 \end{aligned}$$

which yields

$$\begin{aligned} \dot{\tilde{e}}_x &= A\tilde{e}_x + (A^{-1}B_1 + B_2)e_2 \\ \dot{\tilde{\chi}} &= S\tilde{\chi} - k_1 S^{-1} G e_1 \\ \dot{e}_1 &= e_2 \\ \dot{e}_2 &= C\tilde{e}_x + \Psi^T\tilde{\chi} - (CA^{-1}B_1 + k_1(1 - \Psi^T S^{-1} G))e_1 - \bar{k}_2 e_2 \ , \end{aligned} \tag{3.13}$$

i.e. a system of the form

$$\dot{x} = Jx$$

with J a matrix having the same structure as the matrix J in the Lemma if we set

$$\begin{aligned} B &= (A^{-1}B_1 + B_2) \\ Q_i &= \Psi_i^T \qquad\qquad\qquad\qquad i = 1, \ldots, r \\ P_i &= -k_1 S_i^{-1} G_i \qquad\qquad\qquad i = 1, \ldots, r \\ Q_0 &= -(CA^{-1}B_1 + k_1(1 - \Psi^T S^{-1} G)) \\ P_0 &= 1 \\ k &= \bar{k}_2 \end{aligned}$$

Thus, conditions $Q_i P_i < 0$, for $i = 1, \ldots, r$, of the Lemma become

$$k_1 \Psi_i^T S_i^{-1} G_i > 0$$

while conditions $QoPo < 0$ of the Lemma becomes

$$CA^{-1}B_1 + k_1(1 - \Psi^T S^{-1} G) = CA^{-1}B_1 + k_1(1 - \sum_{i=1}^{r} \Psi_i^T S_i^{-1} G_i) > 0 \ .$$

Indeed, if G is such that (3.9) and (3.10) hold, the conditions of the Lemma can be met for large $k_1 > 0$ and, consequently, for large $\bar{k}_2 > 0$ the matrix J is Hurwitz.

3.2 A Numerical Example of Take-Off and Landing

In this section we consider some specific numerical examples. In our first example we consider the case in which there is no z, i.e. no zero dynamics, and that $D_1 = D_2 - 0$, $K = 1$ so that the plant reduces to the double integrator. For the second example we consider a one dimensional stable zero dynamics.

For both examples we assume that disturbance d is given as a sum of sinusoids with two different frequencies,

$$d(t) = M_1 \sin(\alpha_1 t + \varphi_1) + M_2 \sin(\alpha_2 t + \varphi_2).$$

In this case we can

$$S_1 = \begin{pmatrix} 0 & 2 \\ -2 & 0 \end{pmatrix}, \qquad S_2 = \begin{pmatrix} 0 & 10 \\ -10 & 0 \end{pmatrix}, \quad S = \begin{pmatrix} S_1 & 0 \\ 0 & S_2 \end{pmatrix}$$

and consider

$$\dot{w} = Sw, \quad w(0) = w_0.$$

Then we chose Ψ_1^T, Ψ_2^T and $\Psi^T = \begin{pmatrix} \Psi_1^T & \Psi_2^T \end{pmatrix}$ so that $d(t) = \Psi^T w(t)$.

Example 3.3: *In our numerical examples we have chosen* $\alpha_1 = 2$, $\alpha_2 = 10$,

$$\Psi_1^T = \begin{pmatrix} 0 & 2 \end{pmatrix}, \qquad \Psi_2^T = \begin{pmatrix} 0 & 2 \end{pmatrix}$$

$$w_0 = \begin{pmatrix} 0 \\ 25 \\ 0 \\ 50 \end{pmatrix}$$

so that

$$d(t) = 50\cos(2t) + 100\cos(10t).$$

Next we choose G_1, G_2, k_1 and k_2 so that the conditions of Proposition 3.1 are satisfied.

In the numerical example we have set $k_1 = 10$ and

$$G_1 = \begin{pmatrix} 0.2 \\ -0.2 \end{pmatrix} \qquad G_2 = \begin{pmatrix} 1 \\ -1 \end{pmatrix}$$

which gives

$$\Psi_1^T S_1^{-1} G_1 = 0.2, \qquad \Psi_2^T S_2^{-1} G_2 = 0.2$$

and

$$\Psi_1^T S_1^{-1} G_1 + \Psi_2^T S_2^{-1} G_2 = .4 < 1$$

so the conditions of Proposition 3.1 are fulfilled.

We proceed to chose $k_2 > 0$ so that the matrix J in (3.11) is Hurwitz. Our choice of k_2 will be based on a root locus design. Since we do not have a z component in this example the matrix J can be reduced by deleting the first column and the first row. Then we set $k_2 = k = 0$ and define the resulting matrix to be J_0, i.e.,

$$J_0 = \begin{pmatrix} S_1 & \cdot & 0 & 0 & P_1 \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & \cdot & S_r & 0 & P_r \\ 0 & \cdot & 0 & 0 & P_0 \\ Q_1 & \cdot & Q_r & Q_0 & 0 \end{pmatrix}$$

Then we define matrices

$$\mathcal{B}^T = (0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 1), \quad \mathcal{C} = (0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 1)$$

and consider the system (J_0, B, C) with feedback law $u = -k2y$ and transfer function

$$\mathcal{C}(sI - J_0)^{-1}\mathcal{B}.$$

Notice that

$$J = J_0 - k_2\mathcal{B}\mathcal{C},$$

and, more importantly, the eigenvalues of J are the closed loop poles of this system. If $k_1 > 0$ our Proposition predicts that for large values of k_2 the matrix J is Hurwitz. For $k_1 = 10$ we plot the locus of the roots of the characteristic polynomial of J, viewing k_2 as a gain parameter. For large k_2, five of the 6 roots should approach five (zeros) on the imaginary axis, at 0, $\pm 2i$, $\pm 10i$, while the 6-th one goes to - ∞.

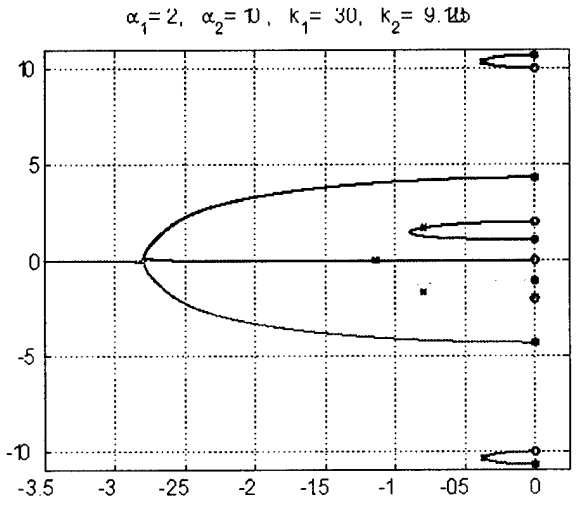
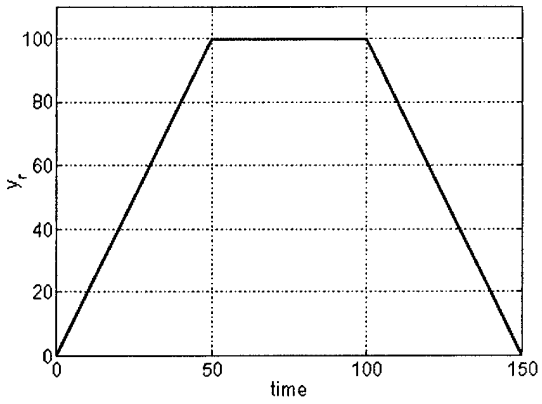
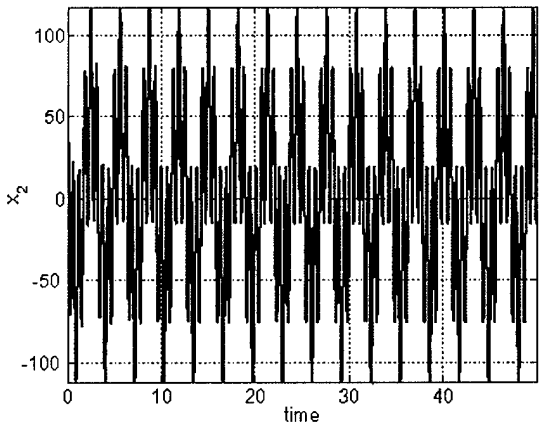


Figure 3-1. Plot of Root Locus

In this example we take a simplified model of aircraft take-off and landing with a reference signal given by

$$y_{\text{ref}}(t) = \begin{cases} 2t, & 0 < t < 50 \\ 100, & 50 < t < 100 \\ 300 - 2t, & 100 < t < 150 \end{cases} .$$

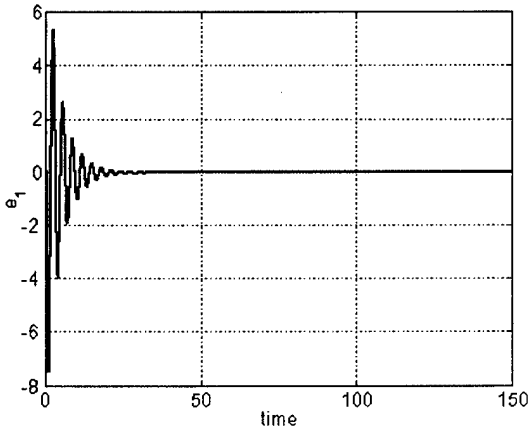


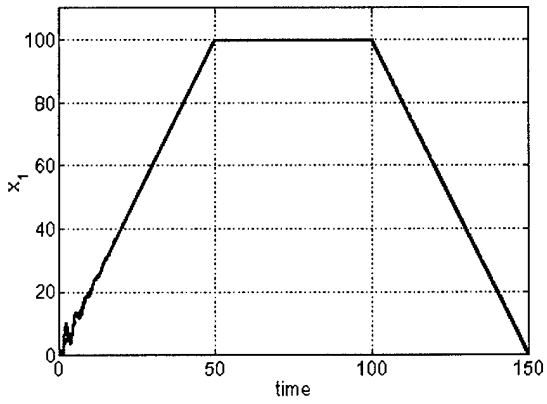


(left) plot of y_{ref} **(right) plot of $x_{2,n} = x_2 + d$**

Figure 3-2. y_{ref} and $x_{2,n}$ in Example 3.3

Notice that y_{rcf}, is not differentiable at t = 50 and t = 100 as it would actually be in practice. Indeed we have not actually used y_{rcf} as given above but rather we have used a Hermite interpolation to round the corners at these points. Since it serves no particular good to produce these more complicated formulas we do not present them here.





(left) plot of $e_1^{(t)} = x_1^{(t)} - y_{ref}^{(t)}$ **(right) plot of x_1 and y_{ref}**

Figure 3-3. e_1, x_1, and y_{ref} in Example 3.3

Example 3.4: In our second numerical example consider a system with a one dimensional stable zero dynamics. In this case we have set A = -1, C = 1, D_1 = 1 and D_2 = 1. The relative values of B_1 and B_2 play a large role in the location of the closed loop poles so we present a "root locus plot" in two different cases: B_1 = 1/4, B_2 = 1/2 and B_1 = 1/2, B_2 = 1/4. Also for this example we compute the reference zero dynamics trajectory z_{rcf} satisfying (3.4) for a given initial condition $z_{rcf}(0) = 5$. Once again we have set $a_1 = 2$, $a_2 = 10$, and we have taken

$$\Psi_1^T = (1 \quad 1), \qquad \Psi_2^T = (1 \quad 1)$$

$$w_0 = \begin{pmatrix} 0 \\ 25 \\ 0 \\ 10 \end{pmatrix}$$

so that

$$d(t) = 25\sqrt{2}\sin(2t + \pi/4) + 10\sqrt{2}\sin(10t + \pi/4).$$

Next we chose G_1, G_2, k_1 and k_2 so that the conditions of Proposition 3.1 are satisfied.

In the numerical example we have set k_1= 20 and

$$G_1 = \begin{pmatrix} 0.2 \\ -0.2 \end{pmatrix} \qquad G_2 = \begin{pmatrix} 1 \\ -1 \end{pmatrix}$$

which gives

and

$$\Psi_1^T S_1^{-1} G_1 + \Psi_2^T S_2^{-1} G_2 = .4 < 1 \quad \Psi_1^T S_1^{-1} G_1 = 0.2, \qquad \Psi_2^T S_2^{-1} G_2 = 0.2$$

so the conditions of Proposition 3.1 are fulfilled.

Just as in the previous example we choose $k_2 > 0$ so that the matrix J in (3.11) is Hurwitz. In the present case more care must be taken in making our choice for k_2 as is seen in the root locus plots in Figure 3-4. Just as in Example 3.1, if we first set $k_2 = k = 0$ in J and define the resulting matrix to be J0, i.e.,

$$J_0 = \begin{pmatrix} A & 0 & \cdot & 0 & 0 & B \\ 0 & S_1 & \cdot & 0 & 0 & P_1 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & 0 & \cdot & S_r & 0 & P_r \\ 0 & 0 & \cdot & 0 & 0 & P_0 \\ C & Q_1 & \cdot & Q_r & Q_0 & 0 \end{pmatrix}$$

Then we define matrices

$$\mathcal{B}^T = (0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 1), \quad \mathcal{C} = (0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 1)$$

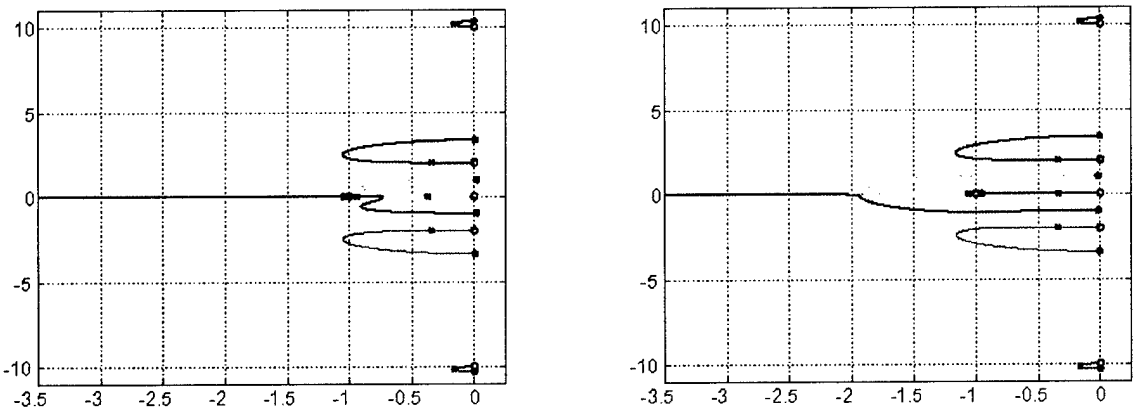
and consider the system (*Jo, B, C*) with feedback law u = -*k2y* and transfer function

$$\mathcal{C}(sI - J_0)^{-1}\mathcal{B}.$$

Notice that

$$J = \mathcal{J}_0 - k_2 \mathcal{B}\mathcal{C},$$

and, more importantly, the eigenvalues of J are the closed loop poles of this system.



(Left) $B = (A^{-1}B_1 + B_2) > 0$, $k_1 = 15$ **(Right) $= (A^{-1}B_1 + B_2) < 0$, $k_1 = 15$**

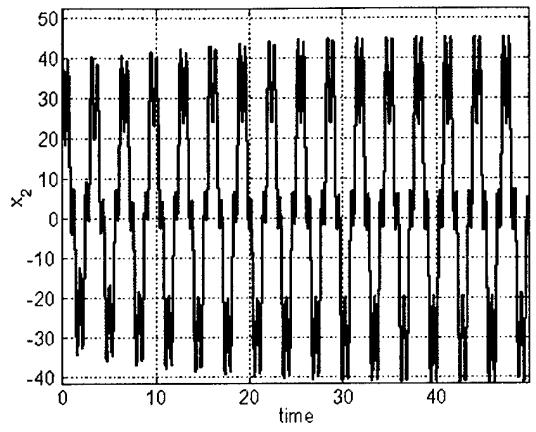
Figure 3-4. Root Locus in Example 3.4

Unlike the first example more care must be exercised in choosing k_1 and k_2. As an example, if we choose $k_1 > 0$, wet $A = -1$, $C = 1$, then the sign of the real parts of the eigenvalues of Jo depend on whether $B = (A^{-1}B_1 + B_2)$ is greater than or less than zero.

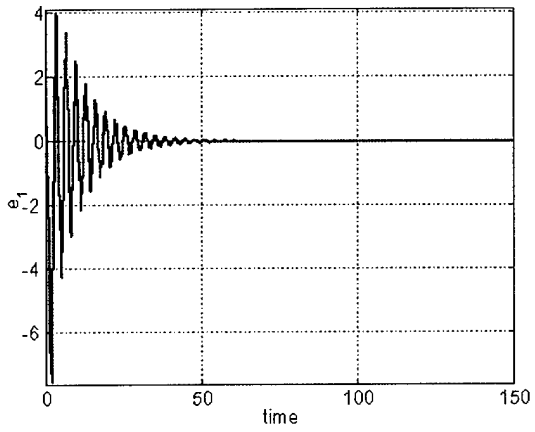
If $k_1 > 0$ our Proposition still predicts that for large values of k_2 the matrix J is Hurwitz. For $k_1 = 15$ and for $B_1 = 1/4$, $B_2 = 1/2$, $B = (A^{-1}B_1 + B_2) > 0$ for which the some of the open loop poles are in the right halfplane while for $k_1 = 15$ and for $B_1 = 1/2$, $B_2 = 1/4$, $B = (A^{-1}B_1 + B_2) < 0$ all open loop poles are in the left half plane for large k_2, five of the 7 closed loop poles approach the five (zeros) on the imaginary axis, at $0, \pm 2i, \pm 10i$, while the 6-th one goes to $-\infty$ and the 7-th approaches the zero at -1.

In this example we take a simplified model of aircraft take-off and landing with a reference signal given by

$$y_{\text{ref}}(t) = \begin{cases} 2t, & 0 < t < 50 \\ 100, & 50 < t < 100 \\ 300 - 2t, & 100 < t < 150 \end{cases}.$$

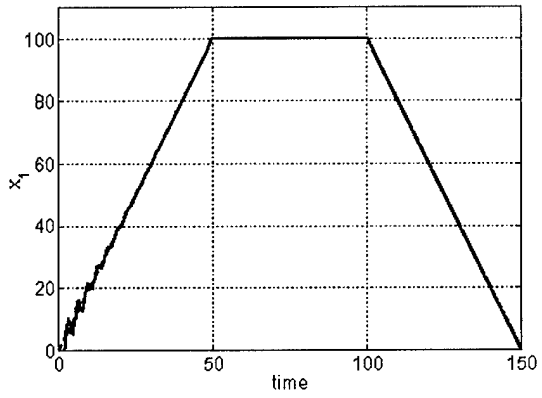


(Left) $x_{2,n} = x_2 + d$

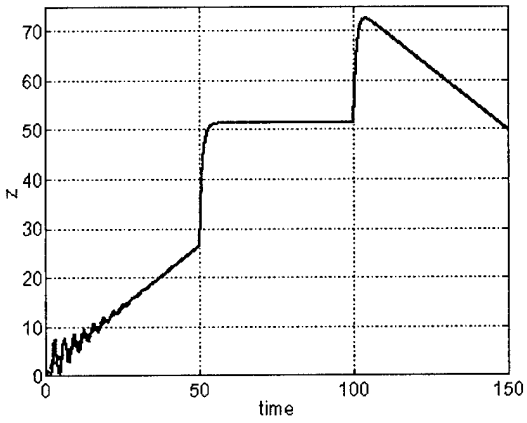


(Right) $e_1(t) = x_1(t) - y_{ref}^{(t)}$

Figure 3-5. $x_{2,n}$ and e_1 in Example 3.4



(Left) x_1 and y_{ref}



(Right) z

Figure 3-6. x_1, y_{ref}, and z in Example 3.4

3.3 Remarks on the Internal Model Principle

One of the remarkable features of the Internal Model Principle is its ability to generate, in a systematic fashion, classical control designs. More importantly, it can also be used in multivariable, nonlinear and infinite-dimensional contents.

As a first illustration of this point, consider a relative degree 1, minimum phase scalar-input scalar-output system

$$\begin{aligned} \dot{z} &= Az + By \\ \dot{y} &= Cz + Dy + ku \end{aligned}$$

where $y_{noisy} = y + d$ with $d(t)$ being an unknown constant. Setting

$$e = y - y_{ref}$$

we obtain the relative degree one analog of (3.3)-(3.6). In this case, the constant disturbance takes the form

$$d = vw$$

where w is the state of the exogenous system

$$\dot{w} = 0.$$

This generates the internal model

$$\dot{\xi} = -Ge$$
$$v = \psi e$$

which is a proportional-integral (PI) controller. Stability of the closed loop system

$$\begin{aligned} \dot{e}_x &= Ae_x + Be \\ \dot{\chi} &= -Ge \\ \dot{e} &= Ce_x + \psi\chi - ke \end{aligned}$$

is guaranteed, whenever $\psi G > 0$ by Lemma 3.2 (see [7]).

As a second illustration, consider the system described in Example 3.1. The disturbance is a sum of two sinusoids having frequencies $a_1 = 2$ and $a_2 = 10$. Figure 3-7 below depicts the Bode magnitude and phase plots of the closed-loop transfer function from the disturbance to the tracking error, obtained by using an internal model based design. The magnitude plot is, of course, that of a typical notch filter having notches (or transmission zeroes) at precisely $2i$ and $10i$, reflecting the fact that the filter absorbs sinusoids at those frequencies.

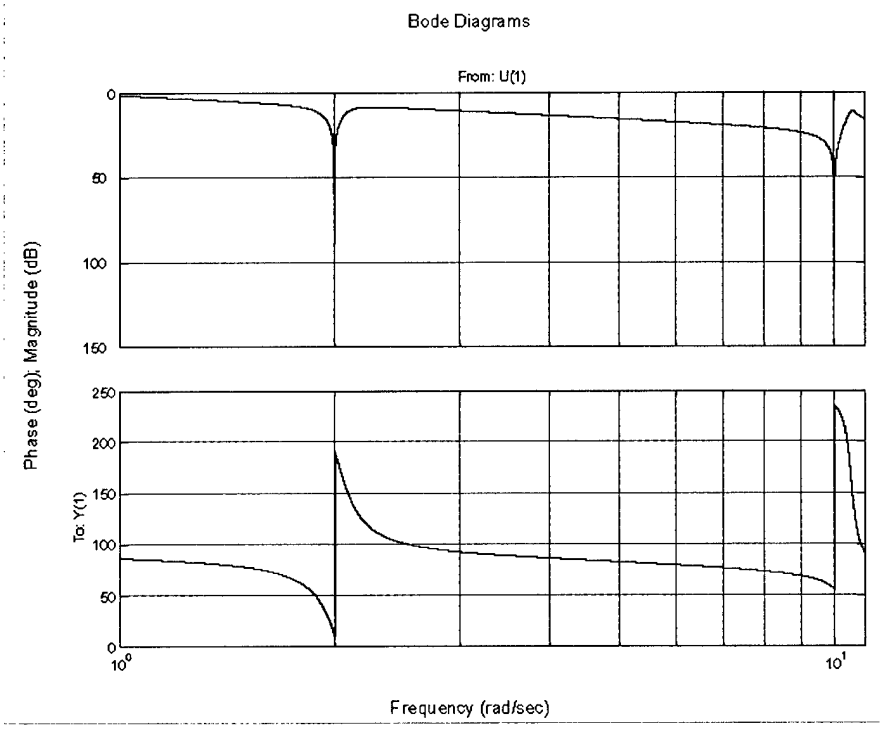


Figure 3-7. Bode Plots of the System in Example 3.1

3.4 The Case of a MIMO System

In the following we briefly sketch how the analysis developed in section 3 can be easily extended to deal also with multi inputs-multi outputs systems. We consider the class of minimum-phase linear systems with three inputs and three outputs with vector relative degree [2, 2, 2]. As is well-known, this class of systems can always be put, after a suitable change of coordinates, in the normal form

$$\begin{aligned} \dot{z} &= Az + B_1 x_1 + B_2 x_2 \\ \dot{x}_1 &= x_2 \\ \dot{x}_2 &= Cz + D_1 x_1 + D_2 x_2 + Ku \\ y &= x_1 \end{aligned} \tag{3.14}$$

where z∈ IRr

$$x_1 = \begin{pmatrix} x_{11} & x_{12} & x_{13} \end{pmatrix}^T \in \mathbb{R}^3 \qquad x_2 = \begin{pmatrix} x_{21} & x_{22} & x_{23} \end{pmatrix}^T \in \mathbb{R}^3 ,$$

$u \in \mathrm{IR}^3$ is the vector of control inputs and $z \in \mathrm{IR}^3$ of controlled outputs, with the matrix A which is Hurwitz and the high frequency gain matrix $K \in \mathrm{IR}^3 \times \mathrm{IR}^3$ which is non singular. Our goal is to force the output vector y to asymptotically track a vector of known reference signals.

$$y_{\mathrm{ref}} = \begin{pmatrix} y_{\mathrm{ref},1} & y_{\mathrm{ref},2} & y_{\mathrm{ref},3} \end{pmatrix}^T$$

with the design of a proportional-derivative output feedback control law. Similarly to section 4 the challenging aspect of this apparently simple tracking problem is given by the presence of additive noise on the velocity measures. More precisely we suppose, as above, that the measure of the position vector x_1 is noise-free, but the measured velocity vector is corrupted by additive harmonic noise, namely

$$x_{2,\mathrm{noisy}}(t) = x_2(t) + d(t) \qquad \text{with} \qquad d(t) = \begin{pmatrix} d_1(t) & d_2(t) & d_3(t) \end{pmatrix}^T$$

where the signals *di, i* = 1,2,3, are given as superposition of sinusoidal functions of time. In particular the signals *di(t), i* = 1, 2, 3, are thought as

$$d_i(t) = \Psi_i^T w_i(t) \qquad \text{with} \qquad \dot{w}_i(t) = S_i w_i(t) \qquad w_i \in \mathbb{R}^{2r_i}$$

where S_i is a block-diagonal square matrix of dimension $2r_i$ which describes a bank of r_i oscillators at different frequencies.

By mimicking the design methodology of the SISO case we define by z_{ref} any solution of

$$\dot{z}_{\mathrm{ref}} = Az_{\mathrm{ref}} + B_1 y_{\mathrm{ref}} + B_2 \dot{y}_{\mathrm{ref}}$$

and we choose the preliminary control law

$$u = K^{-1}\left(-Cz_{\mathrm{ref}} - D_1 x_1 - D_2 \dot{y}_{\mathrm{ref}} + \ddot{y}_{\mathrm{ref}} + v\right) \tag{3.15}$$

where *v* is a vector of residual control inputs. Defining

$$\begin{aligned} e_1 &= y - y_{\mathrm{ref}} &&= x_1 - y_{\mathrm{ref}} \\ e_2 &= \dot{y} - \dot{y}_{\mathrm{ref}} &&= x_2 - \dot{y}_{\mathrm{ref}} \end{aligned}$$

and $z = z - z_{ref}$, simple computations show that system (3.4) under the feedback (3.15), in the new error coordinates, reads as

$$\begin{aligned} \dot{\tilde{z}} &= A\tilde{z} + B_1 e_1 + B_2 e_2 \\ \dot{e}_1 &= e_2 \\ \dot{e}_2 &= C\tilde{z} + D_2 e_2 + v \end{aligned} \tag{3.16}$$

The design of the vector of inputs ν able to asymptotically stabilize the system (3.16), and hence to solve the tracking problem can be easily carried out by means of a three-steps procedure in which, at each step, just one control variable v_i designed, following exactly the design methodology illustrated in the SISO case.

To sketch how this procedure can be carried out consider, for the first control input v_1, the following dynamic law

$$\begin{aligned} \dot{\xi}_1 &= S_1\xi_1 - G_1 k_{11} e_{11} \\ v_1 &= -k_{11}(x_{11} - y_{\mathrm{ref},1}) - k_{21}(x_{21,\mathrm{noisy}} - \dot{y}_{\mathrm{ref},1}) + \Psi_1\xi_1 \\ &= -k_{11}e_{11} - k_{21}e_{21} + \Psi_1\xi_1 - k_{21}\Psi_1 w_1 \end{aligned} \tag{3.17}$$

in which k_{11}, k_{21} and the entries of G_1 are design parameters. Simple computations show that, having defined

$$\chi_1 = \xi_1 - k_{21} w_1 ,$$

system (3.16) with the partial control law (3.17) can be rewritten as

$$\begin{aligned} \dot{\bar{z}} &= \bar{A}\bar{z} + \bar{B}_1\bar{e}_1 + \bar{B}_2\bar{e}_2 \\ \dot{\bar{e}}_1 &= \bar{e}_2 \\ \dot{\bar{e}}_2 &= \bar{C}\bar{z} + \bar{D}_2\bar{e}_2 + \bar{v} \end{aligned} \tag{3.18}$$

where

$$\bar{z} = \begin{pmatrix} \tilde{z} & \chi_1 & e_{11} & e_{21} \end{pmatrix}^T \qquad \bar{e}_i = \begin{pmatrix} e_{i,2} & e_{i,3} \end{pmatrix}^T \qquad \bar{v} = \begin{pmatrix} v_2 & v_3 \end{pmatrix}^T$$

the matrix $\overline{A}$ is defined as

$$\overline{A} = \begin{pmatrix} A & 0 & B_1' & B_2' \\ 0 & S_1 & -G_1 k_{11} & 0 \\ 0 & 0 & 0 & 1 \\ C' & \Psi_1^T & -k_{11} & -(k_{21} - D_2') \end{pmatrix} \tag{3.19}$$

and $\overline{B}_1, \overline{B}_2, \overline{D}_1, \overline{D}_2, B_1', B_2', C', D_2'$ are suitably defined matrices

Now it can be easily realized that it is possible to chose k_{11}, k_{21}, and G_1 so that the matrix $\overline{A}$ is Hurwitz. To this end just note that the matrix characterizing system (3.8) has the same structure of (3.19). Hence, by Proposition 3.1, we can claim the existence of a vector G_1 such that for suitably large k_{11} and k_{22}, the matrix $\overline{A}$ is Hurwitz.

From this, once $G1$, k_{11} and k_{22} have been fixed, the procedure can be iterated in order to design v_2 and v_3. As a matter of act the system (3.18) exhibits the same structure of the original

error system (3.16), with the state variables $\bar{e}_i$, an $\bar{z}$ which replace e_i and $\tilde{z}$, with the vector of control input $\bar{v}$ which replaces v, and with the Hurwitz matrix $\bar{A}$ which replaces A. This allows us to design also v_2, and in the third step v_3, using the same procedure used for the SISO case and come out with a proportional-derivative stabilizer of the form

$$\begin{aligned} \dot{\xi} &= S\xi - GK_1e_1 \\ v &= -K_1e_1 - K_2e_2 + \Psi\xi \end{aligned} \tag{3.20}$$

with

$$S = \begin{pmatrix} S_1 & 0 & 0 \\ 0 & S_2 & 0 \\ 0 & 0 & S_3 \end{pmatrix} \qquad K_1 = \begin{pmatrix} k_{11} & 0 & 0 \\ 0 & k_{12} & 0 \\ 0 & 0 & k_{13} \end{pmatrix} \qquad K_2 = \begin{pmatrix} k_{21} & 0 & 0 \\ 0 & k_{22} & 0 \\ 0 & 0 & k_{23} \end{pmatrix}$$

$$G = \begin{pmatrix} G_1 & 0 & 0 \\ 0 & G_2 & 0 \\ 0 & 0 & G_3 \end{pmatrix} \qquad \Psi = \begin{pmatrix} \Psi_1 & 0 & 0 \\ 0 & \Psi_2 & 0 \\ 0 & 0 & \Psi_3 \end{pmatrix}.$$

Hence the overall control law is obtained from the composition of (3.15) and (3.20)

Example 3.5:

In our 3-dimensional numerical examples we have chosen $a_1 = 2$, $a_2 = 3$, $a_3 = 4$, $\Psi_1^T = \Psi_2^T = \Psi_3^T = (2 \quad 2)$,

$$w_0 = \begin{pmatrix} 0 \\ 25 \\ 0 \\ 50 \end{pmatrix}$$

so that

$$d(t) = \begin{bmatrix} 5(\sin(\alpha_1 t) + \cos(\alpha_1 t)) \\ 10(\sin(\alpha_2 t) + \cos(\alpha_2 t)) \\ 5(\sin(\alpha_3 t) + \cos(\alpha_3 t)) \end{bmatrix}.$$

Next, we choose G_1, G_2, G_3 k_1 and k_2 so that the conditions of Proposition 3.1 are satisfied. We have also set $D_1 = D_2 = I_{3x3}$.

In the numerical example we have set $k_1 = 10$ and

$$G_1 = \begin{pmatrix} 0.2 \\ -0.2 \end{pmatrix}, \qquad G_2 = \begin{pmatrix} 1 \\ -1 \end{pmatrix}, \qquad G_3 = \begin{pmatrix} 1 \\ -1 \end{pmatrix}$$

which gives

$$\Psi_1^T S_1^{-1} G_1 = 1/5, \qquad \Psi_2^T S_2^{-1} G_2 = 2/3, \qquad \Psi_3^T S_3^{-1} G_3 = 1/10$$

and

$$\Psi_1^T S_1^{-1} G_1 + \Psi_2^T S_2^{-1} G_2 + \Psi_3^T S_3^{-1} G_3 = \frac{87}{100} < 1$$

so the conditions of Proposition 3.1 are fulfilled.

From the theory developed in Section 3.1 we see that the closed loop system can be written as $\dot{\mathrm{x}} = \mathrm{JX}$ where

$$\mathbf{X} = \begin{bmatrix} \xi \\ e_1 \\ e_2 \end{bmatrix} \in \mathbb{R}^{12},$$

and, just as in Example 3.3, since there is no z component, the matrix **J** can be reduced by deleting the first column and first row, to obtain

$$J = \begin{bmatrix} S & -GK_1 & 0_{6\times 3} \\ 0_{3\times 6} & 0_{3\times 3} & I_{3\times 3} \\ \Psi & -K_1 & -(K_2 - D_2) \end{bmatrix}.$$

Since this is a MIMO example we cannot directly appeal to root locus methods but if we set

$$K_2 = k_2 I_{3\times 3}, \quad \text{and set } k = (k_2 - 1)$$

then it is still simple to compute a "root locus type plot" of the closed loop poles as functions of the scalar parameter $k > 0$.

Namely, we define the matrix to be $\mathbf{J}_0$ by

$$\mathrm{J}_0 = \begin{bmatrix} S & -GK_1 & 0_{6\times 3} \\ 0_{3\times 6} & 0_{3\times 3} & I_{3\times 3} \\ \Psi & -K_1 & 0_{3\times 3} \end{bmatrix}.$$

Then we define matrices

$$B^T = \begin{pmatrix} 0_{1\times 9} & 1 & 1 & 1 \end{pmatrix}, \quad C = \begin{pmatrix} 0_{1\times 9} & 1 & 1 & 1 \end{pmatrix}$$

so that

$$\mathrm{J} = \mathrm{J}_0 - kBC$$

and consider the eigenvalues of **J** as the closed loop poles of this system.

For $k_1 = 10$ we plot the locus of the roots of the characteristic polynomial of **J**, viewing k_2 (or k) as a gain parameter.

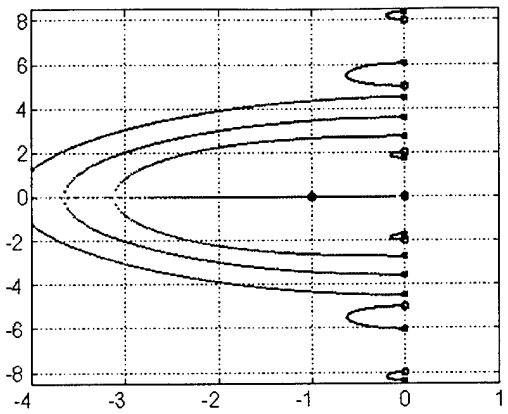


Figure 3-8. Plot of Root Locus

In this example we track a three dimensional reference trajectory

$$y^{\mathrm{ref}}(t) = \left[-10\cos\left(\frac{3\pi t}{100}\right) \quad 10\sin\left(\frac{3\pi t}{100}\right) \quad y_3^{\mathrm{ref}}(t) \right]^T$$

where

$$y_3^{\mathrm{ref}}(t) = \begin{cases} t, & 0 < t < 40 \\ 40 + 5\sin\left(\dfrac{3\pi(t-40)}{60}\right), & 40 < t < 100 \\ -\frac{4}{5}(t-150), & 100 < t < 150 \end{cases}.$$

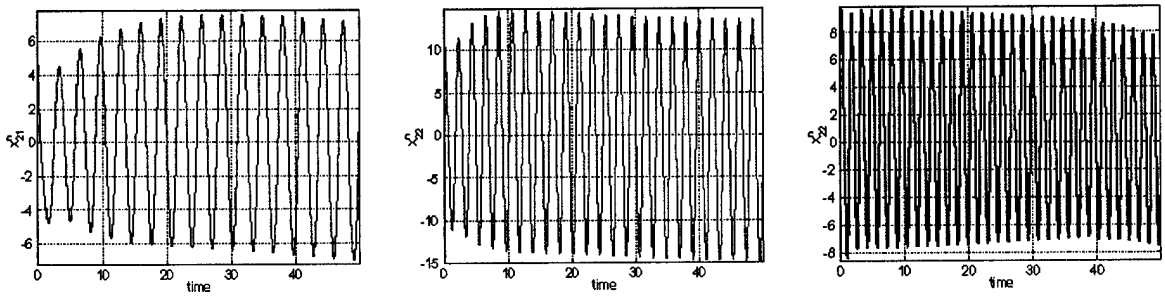


Figure 3-9. $x_{21,n} = x_{21} + d_1$, $x_{22,n} = x_{22} + d_2$, $x_{23,n} = x_{23} + d_3$

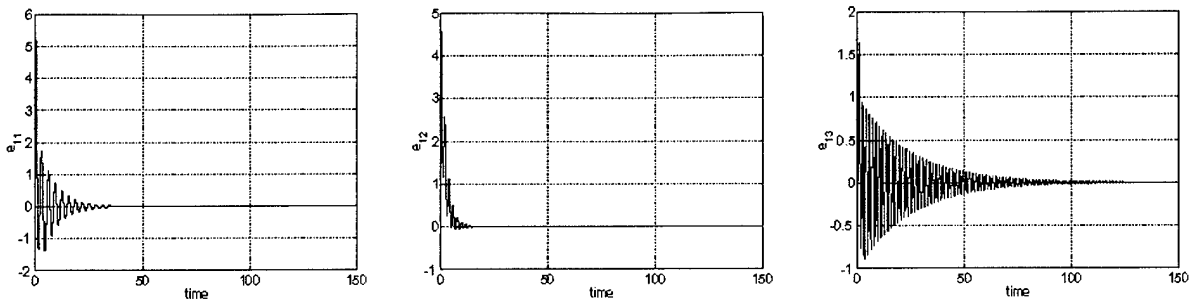


Figure 3-10. $e_{1j}(t) = x_{1j}(t) - yj_{,}^{ref}(t)$ for j = 1, 2, 3

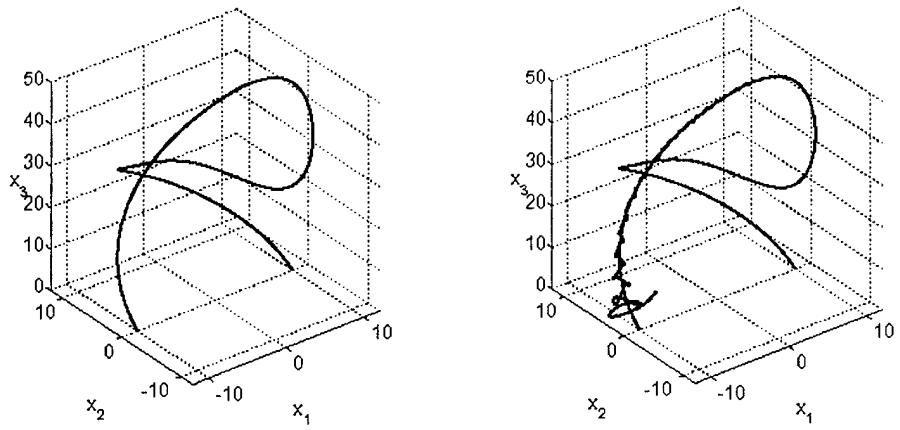


Figure 3-11. y^{ref}, x_1 and y^{ref}

4.0 ROBUST ADAPTIVE NONLINEAR OUTPUT REGULATION
- An Application To UCAV Flight Control -

In this section, we discuss an application of Output Regulation methods to UCAV flight control. For this purpose, Neural Networks and Internal Model Principle Nonlinear Output Regulation (IMPNOR) methods are considered. These two methods are applied to the UCAV X-45A linear model independently, and the results are compared. We start with the plant model.

4.1 Plant Model

Beginning in this section, adaptive model reference augmentation of a baseline controller is introduced using the UCAV X-45A model, as an example. It should be noted, that the method is not restricted to this particular model, and that it is applicable to any plant dynamics that is taken to be of the form:

$$\dot{x}_p = A_p x_p + \underbrace{B G}_{B_p} \Lambda\left(\delta + K_0\left(x_p\right)\right) = A_p x_p + B_p \Lambda\left(\delta + K_0\left(x_p\right)\right) \tag{4.1}$$

In (4.1), using the UCAV X-45A flight dynamics nomenclature,

$$x_p = \left(\alpha \quad \beta \quad p \quad q \quad r\right)^T \tag{4.2}$$

is the $n_p = 5$ – dimensional state vector that consists of angle of attack α, angle of sideslip β, body roll rate p, body pitch rate q, and body yaw rate r. The vector of *virtual* controls δ is $(m\times 1)$ – dimensional with $m = 3$, G is the $(M\times m)$ – control allocation matrix, Λ is the $(m\times m)$ – *unknown* constant *diagonal* matrix with positive diagonal elements, and

$$G\Lambda\delta = \left(\delta_{LOB} \quad \delta_{LMB} \quad \delta_{LIB} \quad \delta_{RIB} \quad \delta_{RMB} \quad \delta_{ROB} \quad \delta_{Tvec}\right)^T \tag{4.3}$$

is the $M = 7$ – dimensional vector of *actual* control inputs, (positions) that consists of 6 elevons and a thrust vectoring. Also in (4.1), A_p is a $(n\times n)$ – constant *known* matrix, B is a $\left(n_p \times M\right)$ – constant *known* matrix, B_p is a $\left(n_p \times m\right)$ – constant *known* matrix, and $K_0\left(x_p\right)$ is the $(m\times 1)$ – *unknown* state-dependent vector. Note that while Λ models unknown control failures, the vector-function $K_0\left(x_p\right)$ represents matched unknown nonlinear effects.

Vertical acceleration a_z (positive down) is computed as the system (4.1) output signal.

$$a_z = C_p x_p + \underbrace{D G}_{D_p} \Lambda\left(\delta + K_0\left(x_p\right)\right) = C_p x_p + D_p \Lambda\left(\delta + K_0\left(x_p\right)\right) \tag{4.4}$$

In addition, it assumed that the entire system state x_p is available for control design purposes.

4.2 Baseline Inner-Loop Controller

Dynamics of the UCAV X-45A baseline inner-loop controller is defined as follows:

$$\dot{x}_c = A_c x_c + B_{1c} x_p + B_{2c} u \tag{4.5}$$

where

$$x_c = \begin{pmatrix} q_I & p_I & r_I & r_w \end{pmatrix}^T \tag{4.6}$$

is the $n_c = 4$ – dimensional controller state vector that consists of pitch integrator q_I, roll integrator p_I, yaw integrator r_I, and yaw rate washout filter signal r_w. The vector of inner-loop commands

$$u = \begin{pmatrix} a_z^{cmd} & \beta^{cmd} & p^{cmd} & r^{cmd} \end{pmatrix}^T \tag{4.7}$$

is comprised of $n_u = 4$ commanded signals: vertical acceleration a_z^{cmd}, sideslip β^{cmd}, roll rate p^{cmd}, and yaw rate r^{cmd}. Combining (4.5) and (4.7) with (4.1), the corresponding augmented plant-controller system becomes:

$$\underbrace{\begin{pmatrix} \dot{x}_p \\ \dot{x}_c \end{pmatrix}}_{\dot{x}} = \underbrace{\begin{pmatrix} A_p & 0 \\ B_{1c} & A_c \end{pmatrix}}_{A} \underbrace{\begin{pmatrix} x_p \\ x_c \end{pmatrix}}_{x} + \underbrace{\begin{pmatrix} B_p \\ 0 \end{pmatrix}}_{B_1} \Lambda\left(\delta + K_0\left(x_p\right)\right) + \underbrace{\begin{pmatrix} 0 \\ B_{2c} \end{pmatrix}}_{B_2} u$$

or equivalently

$$\dot{x} = A x + B_1 \Lambda\left(\delta + K_0\left(x_p\right)\right) + B_2 u \tag{4.8}$$

Feedback/feedforward gains for the UCAV X-45A baseline (nominal) inner-loop controller are designed assuming no modeling uncertainties, $\Lambda = I_m$, $K_0\left(x_p\right) = 0_{m\times 1}$, and using the LQR method with projection [16, 17, 18, 19, 20]. The corresponding inner-loop control system takes the form:

$$\delta_L = K_x^T x + K_u^T u \tag{4.9}$$

where K_x and K_u denote the $(n\times m)$ and $(n_u \times m)$ – nominal feedback and feedforward gain matrices, correspondingly. The number of virtual controls is $m = 3$ and $n = n_p + n_c$ represents the order of the extended system (4.8).

Nominal inner-loop feedback (4.9), when applied to the ideal (i.e., without uncertainties) model (4.1), naturally yields the desired (reference) dynamics.

$$\dot{x}_{ref} = \underbrace{\left(A + B_1 K_x^T\right)}_{A_{ref}} x_{ref} + \underbrace{\left(B_2 + B_1 K_u^T\right)}_{B_{ref}} u = A_{ref} x_{ref} + B_{ref} u \tag{4.10}$$

Assumption 1: Reference model matrix A_{ref} is Hurwitz, (i.e., stable).

The inner-loop control objective is to design state feedback control δ for (4.8) such that despite the system uncertainties Λ and $K_0(x_p)$, all closed-loop signals are bounded and the system state x tracks the reference model state x_{ref}.

To this end, it should be noted that the reference model (4.10) was written using the baseline feedback gains designed for the nominal (i.e., no uncertainties) X-45A model. In order to achieve the inner-loop control objective for the system (4.8) in the presence of modeling uncertainties and unknown control failures, a direct adaptive model following control architecture will be designed.

4.3 Inner-Loop Tracking Error Dynamics

4.3.1 Internal Model-Based Controller

Now define the tracking error to be regulated as

$$e = x - x_{ref}$$

and seek an observable (in the first approximation) controller with state ξ and output κ of the form

$$\begin{aligned}\dot{\xi} &= \eta\ (\xi, e, u_{cmd})\\ \delta &= \kappa\ (\xi, e)\end{aligned}$$

such that $\lim_{t\to\infty} e(t) = 0$ and ξ remains bounded. From Eqs. (4.8) and (4.10)

$$\begin{aligned}\dot{e} &= A_0(\Lambda)e + B_0(\Lambda)x_{ref} + B_1\Lambda[\delta + K_0(e + x_{ref})] + B_u(\Lambda)u_{cmd}\\ \dot{\xi} &= \eta\ (\xi, e, u_{cmd})\\ \delta &= \kappa\ (\xi, e)\end{aligned}$$

where $A_0(\Lambda)=A+B_1\,\Lambda\,K^T_x$, $B_0(\Lambda)= B_1\,(\Lambda\text{-}I)\,K^T_x$, $B_u(\Lambda)= B_1\,(\Lambda\text{-}I)\,K^T_u$, and the control $\delta = \kappa(\xi, e)$ is chosen such that the interconnected dynamics are Hurwitz in the first approximation. In addition, the control scheme must ensure that the nonlinear pitch break dynamics do not perturb the system away from some bounded set near $e = 0$ for failures under consideration. The choice of internal models is

$$\begin{aligned}\dot{\xi}_i &= A_{ref}\xi_i + B_{ref}u_{cmd} + N_i e\\ \delta_i &= \Gamma_i\xi_i + K_{1i}e + K_{2i}\,\mathrm{sgn}(e_1)e_1^{\,2} \qquad i = 1,2,3\\ \delta &= [\delta_1 \quad \delta_2 \quad \delta_3]^T\end{aligned}$$

where the nonlinear conditioning term is chosen so as to bound pitch-break dynamics based on the knowledge that the pitch-break phenomenon is quadratic in angle of attack. The gain-matrix entries on the nonlinear error feedback terms are chosen large enough to bound the pitch-break dynamics, but not so large that excessive control effort is produced. The form shown

$$K_{2i}\,\mathrm{sgn}(e_1)e_1^{\,2}$$

gives a 'cubic' stability benefit with a 'quadratic-in-the-error' time-varying magnitude. The linear error-feedback gain-matrices K_{1i} and N_i are chosen to make A_0 Hurwitz and to make the domain of attraction of the error large enough to bound coupled error-reference terms that are introduced by the quadratic pitch-break dynamics. The role of $\Gamma_i\xi_i$ is to approximate the residual purely reference terms. This controller gives bounded error tracking for the closed inner-loop dynamics for the failure cases considered. The outer-loop problem was not considered, although the inner-loop control is sufficient to keep the outer-loop dynamics well-contained for inner-loop commands.

The size of the neighborhood can be determined via Lyapunov analysis. Note that this controller yields approximation regulation because the pitch-break phenomenon introduces a term that is proportional to x^2_{ref} that is not directly accounted for in the design of the IMPNOR scheme. However, this term does not adversely affect the steady-state solution, and, consequently, was not taken into account. Although, it is possible to realize a higher order IMPNOR scheme that contains a copy of the dynamics associated with this quadratic reference signal. Simulation results are provided in Section 4.6.2 for the Inner Loop and in Section 4.9.2 for the Outer Loop.

4.3.2 Adaptive Control Set-up

Inner-loop tracking performance can be quantified by the corresponding tracking error signal:

$$e = x - x_{ref} \tag{4.11}$$

Similar to [13, 14, 15], direct adaptive control input is formed:

$$\delta = \hat{K}_x^T x + \hat{K}_u^T u - \hat{K}_0\left(x_p\right) \tag{4.12}$$

In (4.12), $\hat{K}_x$, $\hat{K}_u$, and $\hat{K}_0\left(x_p\right)$ represent the adaptive feedback/feedforward matrix gains that will be determined through the Lyapunov stability analysis.

Using (4.12), the corresponding closed-loop system dynamics becomes:

$$\begin{aligned}\dot{x} &= A x + B_1 \Lambda\left(\hat{K}_x^T x + \hat{K}_u^T u - \hat{K}_0\left(x_p\right) + K_0\left(x_p\right)\right) + B_2 u \\ &= \underbrace{\left(A + B_1 \Lambda \hat{K}_x^T\right)}_{\hat{A}} x + \underbrace{\left(B_2 + B_1 \Lambda \hat{K}_u^T\right)}_{\hat{B}} u + B_1 \Lambda\left(-\hat{K}_0\left(x_p\right) + K_0\left(x_p\right)\right)\end{aligned} \tag{4.13}$$

Comparing system (4.13) with the reference model dynamics (4.10), the following assumptions are stated.

Assumption 2: For a constant unknown diagonal matrix Λ and a constant control allocation matrix G, there exist "true" gains K_x^*, K_u^* such that the so-called *uncertainty matching conditions* take place:

$$\begin{cases} A_{ref} = A + B_1 \Lambda\left(K_x^*\right)^T \\ B_{ref} = B_2 + B_1 \Lambda\left(K_u^*\right)^T \end{cases} \tag{4.14}$$

Remark 1: Knowledge of the true parameters $K_{\bar{x}}^*$, $K_{\bar{u}}^*$ will not be required for adaptive control design. Only their existence is assumed.

Inner-loop tracking error dynamics is obtained by subtracting (4.10) from (4.13).

$$\begin{aligned}\dot{e} &= \hat{A}x - A_{ref}\,x_{ref} + \left(\hat{B} - B_{ref}\right)\,u - B_1\,\Lambda\left(\underbrace{\hat{K}_0\left(x_p\right) - K_0\left(x_p\right)}_{\Delta K_0\left(x_p\right)}\right) \\ &= A_{ref}\,e + \left(\hat{A} - A_{ref}\right)x + \left(\hat{B} - B_{ref}\right)u - B_1\,\Lambda\,\Delta K_0\left(x_p\right)\end{aligned} \tag{4.15}$$

Using uncertainty-matching conditions (4.14), the tracking error dynamics becomes:

$$\begin{aligned}\dot{e} &= A_{ref}\,e + B_1\,\Lambda\left(\underbrace{\left(K_x - K_x^*\right)^T}_{\Delta K_x}\,x + \underbrace{\left(K_u - K_u^*\right)^T}_{\Delta K_u}\,u - \Delta K_0\left(x_p\right)\right) \\ &= A_{ref}\,e + B_1\,\Lambda\left(\Delta K_x^T\,x + \Delta K_u^T\,u - \Delta K_0\left(x_p\right)\right)\end{aligned} \tag{4.16}$$

To this end, we approximate the uncertainty $K_0\left(x_p\right)$ using multi-input-multi-output feedforward neural network (NN) with N_0 radial basis function (RBF) neurons in its inner layer, [22, 23]. The network computes m linear combinations of a suitably chosen set of radial basis functions $\left\{\varphi_j\left(x_p\right)\right\}_{j=1}^{N_0}$, [24].

$$\hat{K}_0\left(x_p\right) = \begin{pmatrix} \sum_{j=1}^{N_0} \hat{\theta}_{j1}\,\varphi_j\left(x_p\right) \\ \cdots \\ \sum_{j=1}^{N_0} \hat{\theta}_{jm}\,\varphi_j\left(x_p\right) \end{pmatrix} = \hat{\Theta}^T\,\Phi\left(x_p\right) \tag{4.17}$$

The RBF NN Universal Approximation Theorem [21] states that given an approximation tolerance $\varepsilon_0^* > 0$, and a compact set $X \subset R^n$, there must exist an integer N_0 and a "true" constant matrix $\Theta \in R^{N_0 \times m}$ such that for all $x_p \in X \subset R^n$:

$$K_0\left(x_p\right) = \Theta^T\,\Phi\left(x_p\right) + \varepsilon_0\left(x_p\right) \tag{4.18}$$

and

$$\left\|\varepsilon_0\left(x_p\right)\right\| \le \varepsilon_0^* \tag{4.19}$$

In other words, given enough neurons, (i.e., RBF functions), one can approximate a nonlinear function to within any accuracy.

Using neural approximation (4.18), we get

$$\Delta K_0(x_p) = \hat{K}_0(x_p) - K_0(x_p) = \underbrace{\left(\hat{\Theta} - \Theta\right)^T}_{\Delta\Theta} \Phi(x_p) - \varepsilon_0(x_p) \tag{4.20}$$

In (4.20), $\Delta\Theta = \hat{\Theta} - \Theta$, and $\varepsilon_0(x_p)$, represent unknown parameter estimation errors, and uncertainty approximation errors, correspondingly. Substituting (4.20) into (4.16) yields:

$$\dot{e} = A_{ref}\, e + B_1 \Lambda\left(\Delta K_x^T\, x + \Delta K_u^T\, u - \Delta\Theta^T \Phi(x_p) + \varepsilon_0(x_p)\right) \tag{4.21}$$

or, equivalently

$$\dot{e} = A_{ref}\, e + B_1 \Lambda\left(\underbrace{\begin{pmatrix}\Delta K_x^T & \Delta K_u^T & \Delta\Theta^T\end{pmatrix}}_{\Delta\Omega^T} \underbrace{\begin{pmatrix} x \\ u \\ -\Phi(x_p)\end{pmatrix}}_{\Psi(x,\,u)} + \varepsilon_0(x_p)\right) \tag{4.22}$$

Hence, the inner-loop tracking dynamics can be written in the form:

$$\dot{e} = A_{ref}\, e + B_1 \Lambda\left(\Delta\Omega^T \Psi(x,u) + \varepsilon_0(x_p)\right) \tag{4.23}$$

4.4 Inner-loop Stability and Adaptation

Without a loss of generality, it is assumed that the uncertain matrix Λ has strictly positive diagonal elements. To this end, consider the following *positive definite* Lyapunov function candidate:

$$V(e, \Delta\Omega) = e^T P e + trace\left(\Delta\Omega^T \Gamma^{-1} \Delta\Omega\, \Lambda\right) \tag{4.24}$$

where *trace* of a matrix is defined as the sum of its diagonal elements. Given a symmetric positive definite matrix Q, let P denote the unique positive definite symmetric solution of the Lyapunov equation:

$$A_{ref}^T P + P A_{ref} = -Q \tag{4.25}$$

Also note that in (4.24) Γ is a positive definite diagonal symmetric matrix.

$$\Gamma = \begin{pmatrix} \Gamma_x & 0 & 0 \\ 0 & \Gamma_u & 0 \\ 0 & 0 & \Gamma_\Theta \end{pmatrix} \tag{4.26}$$

Computing time derivative of the Lyapunov function candidate V along the system (4.23) trajectories, we get:

$$\begin{aligned}\dot{V}(e,\Delta\Omega) &= \dot{e}^T P e + e^T P \dot{e} + 2\,trace\left(\Delta\Omega^T \Gamma^{-1} \dot{\Omega} \Lambda\right) \\ &= \left(A_{ref}\, e + B_1 \Lambda\left(\Delta\Omega^T \Psi(x,u) + \varepsilon_0(x_p)\right)\right)^T P e \\ &+ e^T P\left(A_{ref}\, e + B_1 \Lambda\left(\Delta\Omega^T \Psi(x,u) + \varepsilon_0(x_p)\right)\right) + 2\,trace\left(\Delta\Omega^T \Gamma^{-1} \dot{\Omega} \Lambda\right) \\ &= -e^T Q e + 2 e^T P B_1 \Lambda\, \Delta\Omega^T \Psi(x,u) + 2\,trace\left(\Delta\Omega^T \Gamma^{-1} \dot{\Omega} \Lambda\right) + 2 e^T P B_1 \Lambda \varepsilon_0(x_p)\end{aligned} \tag{4.27}$$

Applying the trace identity $x^T y = trace\left(y x^T\right)$ to the second term in the right hand side of (4.27), yields:

$$\begin{aligned}\dot{V}(e,\Delta\Omega) &= -e^T Q e + 2 e^T P B_1 \Lambda \varepsilon_0(x_p) \\ &+ 2\,trace\left(\Delta\Omega^T \Psi(x,u) e^T P B_1 \Lambda\right) + 2\,trace\left(\Delta\Omega^T \Gamma^{-1} \dot{\Omega} \Lambda\right) \\ &= -e^T Q e + 2 e^T P B_1 \Lambda \varepsilon_0(x_p) + 2\,trace\left(\Delta\Omega^T \left(\Psi(x,u) e^T P B_1 + \Gamma^{-1} \dot{\Omega}\right)\Lambda\right)\end{aligned} \tag{4.28}$$

Based on (4.28), we choose adaptive laws:

$$\dot{\Omega} = \Gamma\, \mathrm{Proj}\left(\Omega, -\Psi(x,u) e^T P B_1\right) \tag{4.29}$$

where $\mathrm{Proj}(\cdot,\cdot)$ denotes the projection operator, [25]. It ensures that the matrix of adaptive parameters Ω does not exceed its pre-specified norm bound $\Omega_{\max}$, and at the same time negative semi-definiteness of the Lyapunov function is maintained. Due to (4.29), an upper bound for the time derivative of the Lyapunov function can be found.

$$\begin{aligned}\dot{V}(e,\Delta\Omega) &\le -e^T Q e + 2 e^T P B_1 \Lambda \varepsilon_0(x_p) \\ &\le -\lambda_{\min}(Q)\|e\|^2 + 2\|e\|\|P B_1\| \max(\Lambda)\varepsilon_0^* \\ &= -\|e\|\left(\lambda_{\min}(Q)\|e\| - 2\|P B_1\| \max(\Lambda)\varepsilon_0^*\right)\end{aligned} \tag{4.30}$$

Consequently, the Lyapunov function derivative $\dot{V}(e,\Delta\Omega)$ becomes semi-negative outside of the compact set:

$$Z = \begin{cases} \|e\| \le \dfrac{2\|P B_1\| \max(\Lambda)\varepsilon_0^*}{\lambda_{\min}(Q)} \\ \|\Omega\| \le \Omega_{\max} \end{cases} \tag{4.31}$$

Thus, adaptation (4.29) yields *bounded tracking*. Moreover, due to the diagonal form of Γ, the adaptive laws can explicitly be written as:

$$\begin{cases} \dot{\hat{K}}_x = \Gamma_x \ \mathrm{Proj}\left(\hat{K}_x, -x e^T P B_1\right) \\ \dot{\hat{K}}_u = \Gamma_u \ \mathrm{Proj}\left(\hat{K}_u, -u e^T P B_1\right) \\ \dot{\Theta} = \Gamma_\Theta \ \mathrm{Proj}\left(\Theta, \Phi\left(x_p\right) e^T P B_1\right) \end{cases} \tag{4.32}$$

4.5 Adaptive Augmentation of the Baseline Inner-Loop Controller

Since the structures of the baseline inner-loop controller (4.9) and of the adaptive feedback (4.12) are similar, the former could be imbedded into the latter.

$$\begin{aligned} \delta &= \delta_L(x,u) + \hat{K}_x^T x + \hat{K}_u^T u - \delta_L(x,u) - \hat{K}_0\left(x_p\right) \\ &= \underbrace{\delta_L(x,u)}_{\delta_L(x,u)} + \underbrace{\left(\hat{K}_x - K_x\right)^T}_{\hat{k}_x^T} x + \underbrace{\left(\hat{K}_u - K_u\right)^T}_{\hat{k}_u^T} u - \underbrace{\hat{K}_0\left(x_p\right)}_{\hat{\Theta}^T \Phi\left(x_p\right)} \\ &= \underbrace{\delta_L\left(x_p, x_c, u\right)}_{\text{nominal}} + \underbrace{\hat{k}_x^T x + \hat{k}_u^T u - \hat{\Theta}^T \Phi\left(x_p\right)}_{\text{adaptive}} \end{aligned} \tag{4.33}$$

In other words, the adaptive system augments the baseline inner-loop controller. In this case, adaptation of the *incremental* feedback gains $\hat{k}_x$, $\hat{k}_u$, $\hat{\Theta}$ starts from zero initial conditions and has the following dynamics:

$$\begin{cases} \dot{\hat{k}}_x = \Gamma_x \ \mathrm{Proj}\left(\hat{k}_x, -x e^T P B_1\right), \quad \hat{k}_x(0) = 0_{n\times m} \\ \dot{\hat{k}}_u = \Gamma_u \ \mathrm{Proj}\left(\hat{k}_u, -u e^T P B_1\right), \quad \hat{k}_u(0) = 0_{n\times p} \\ \dot{\hat{\Theta}} = \Gamma_\Theta \ \mathrm{Proj}\left(\hat{\Theta}, \Phi\left(x_p\right) e^T P B_1\right), \quad \hat{\Theta}(0) = 0_{N_0\times m} \end{cases} \tag{4.34}$$

Note that in (4.34), the projection operator depends on two vector arguments and is defined column-wise:

$$\mathrm{Proj}(\theta, y) = \begin{cases} y - \dfrac{\nabla f(\theta)\left(\nabla f(\theta)\right)^T}{\left\|\nabla f(\theta)\right\|^2} y f(\theta), & \text{if } \left\{ f(\theta) > 0 \text{ and } y^T \nabla f(\theta) > 0 \right\} \\ y, & \text{if not} \end{cases} \tag{4.35}$$

The corresponding convex cost function and its gradient are given below:

$$\begin{cases} f(y) = \|y\|^2 = \dfrac{y^T y - y_{\max}^2}{\varepsilon_y} \\ \nabla f(y) = \dfrac{2}{\varepsilon_y} y \end{cases} \tag{4.36}$$

In (4.36) $y_{\max}$ denotes the norm upper bound imposed on the parameter vector y, while ε_y defines the projection tolerance. Detailed description of the projection operator is given in Appendix A.

Designed to yield bounded tracking, the adaptation process (4.34) has the ability to not only change/reconfigure the nominal feedback gains K_x and K_u, but also to identify and counteract the nonlinear matched uncertainty $K_0(x_p)$ in the system (4.1). As a result, bounded tracking of the reference model trajectories is achieved in spite of unknown control failures and unmodeled nonlinear dynamical effects. The corresponding closed-inner-loop system block-diagram is shown Figure 4-1.

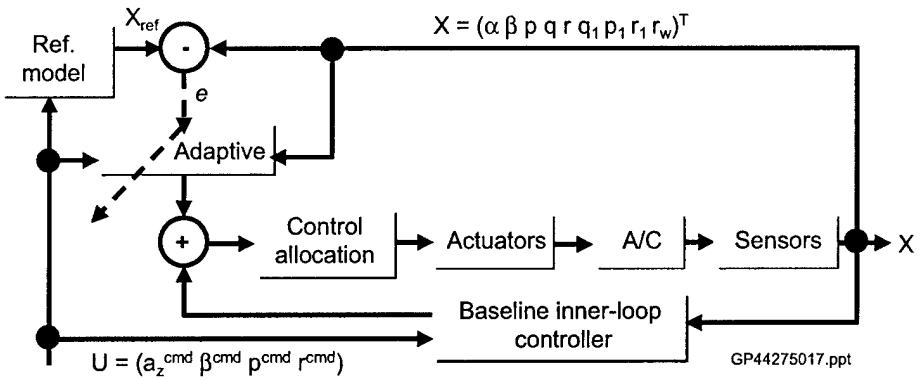


Figure 4-1. Closed-Inner-Loop System Block Diagram

To this end, it should be pointed out that the adaptive system (4.34) was derived using RBF NN approximation (4.18). An alternative approach that uses Single-Hidden-Layer (SHL) feedforward neural networks with sigmoidal activation units instead of the radial basis functions is presented in Appendix B. The two neural networks, RBF NN and SHL NN, are complimentary in their approximation abilities [21].

4.6 UCAV X-45A Inner-Loop Design and Evaluation

4.6.1 Neural Networks-Based Controller

The adaptive augmentation approach illustrated in Figure 4-1 was utilized for the design of the UCAV X-45A adaptive reconfigurable inner-loop controller. For the design purposes, open-loop X-45A aircraft data (i.e., plant) were taken to represent low speed dynamics at Mach = .3, Altitude = 5K ft, and angle of attack AOA = 6 degrees.

Baseline inner-loop feedback / feedforward gains in (4.9) were designed using the LQR method with output projection, [17, 18, 19, 20, 25].

Using relation (4.10), the reference model matrices A_{ref} and B_{ref} were computed. It was verified that the reference model matrix A_{ref} was Hurwitz, (i.e., Assumption 1 was satisfied).

While the 1st and the 3rd components of the modeling uncertainty $K_0(x_p)$ were set to zero, its 2nd component was chosen to represent the so-called "Pitch Break phenomenon" that depended solely on angle of attack, (AOA). Figure 4-2 shows the corresponding data.

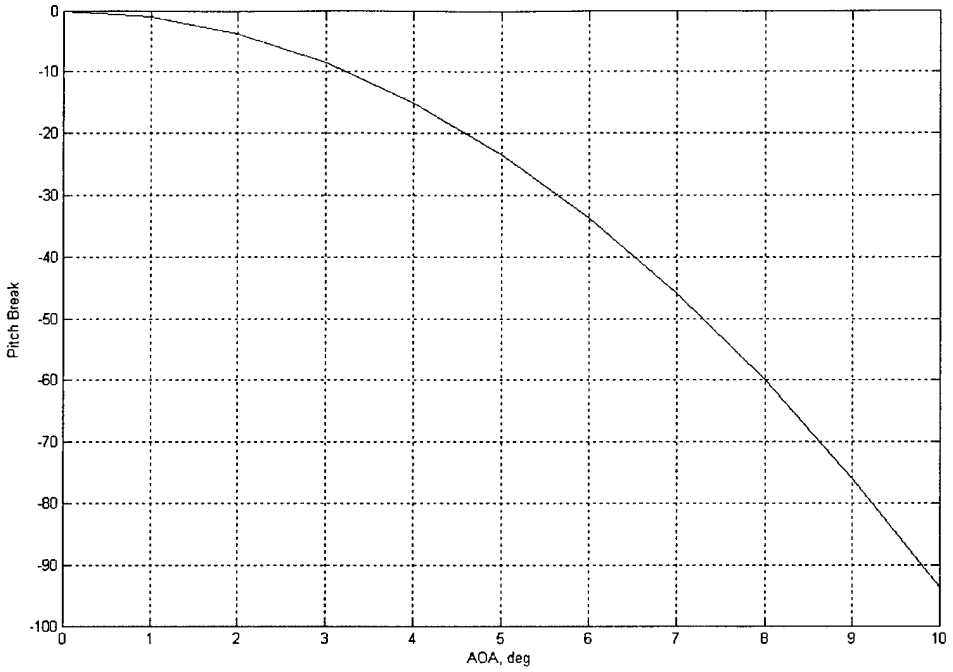


Figure 4-2. Pitch Break vs. AOA data

Radial basis functions were chosen in the form of Gaussians:

$$\varphi_j\left(x_p\right)=\exp\left(-\frac{\left(\alpha-\alpha_j\right)^2+\beta^2+p^2+q^2+r^2}{\sigma^2}\right)$$

AOA break points $\left\{\alpha_j\right\}_{j=1,\ldots,21}$ were spaced evenly between –5 and 5 degrees, and half a degree apart from each other. The Gaussian widths were set to $\sigma=0.5$. The latter provided for a reasonable overlap between the individual basis functions. Also, a unity basis function $\varphi_{22}\left(x_p\right)=1$ was added to the RBF NN in (4.17), $\left(N_0=22\right)$.

The adaptation rates in (4.32) were chosen to be:

$$\begin{cases}\Gamma_x=100\,\mathrm{diag}\left(\begin{bmatrix}0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0\end{bmatrix}\right)\\ \Gamma_u=0\\ \Gamma_\Theta=100\,\mathrm{eye}\left(N_0\right)\end{cases}$$

where "diag" and "eye" denoted diagonal and identity matrices, correspondingly. Note, that in general the Lyapunov function in (4.24) requires nonsingular matrices of adaptation rates. However, since the inner-loop dynamics primarily depends on body angular rates, the rates of adaptation chosen above are adequate, and the rest of the system dynamics can be treated as a modeling uncertainty that is approximated and cancelled on-line by the RBF NN.

Positive semi-definite solution of the Lyapunov equation (4.25) was obtained by choosing

$$Q=\mathrm{diag}\left(\begin{bmatrix}0 & 0 & 2.5 & 2.5 & 2.5 & 0 & 0 & 0 & 0\end{bmatrix}\right)$$

In order to constrain the inner-loop adaptation parameter values in (4.34), column-wise norm upper bounds $y_{\max}$ in (4.36) were set to unity, and projection tolerances ε_y were chosen to be 10% of the norm upper bounds.

Closed-inner-loop system performance was evaluated in the UCAV X-45A flight simulation environment that included 1st order actuator models with a time constant as well as CrowMix – the X-45A nonlinear control allocation logic. Various control failure scenarios with and without the pitch break phenomena were tested. Simulated performance with adaptation was compared to the performance provided by the baseline inner-loop system without the adaptation. The simulation evaluation results are shown in Figures 4-3, 4-4, and 4-5. For comparison purposes, simulation data were obtained from the following three closed-inner-loop systems: a) Adaptation OFF / Failures OFF (blue), b) Adaptation OFF / Failures ON (red), and c) Adaptation ON / Failures ON (black).

Figures 4-3a, 4-3b, and 4-3c demonstrate adaptation benefits when the right outboard (ROB) elevon fails at 1 second into the maneuver. Figure 4-3a indicates that in spite of the *unknown* control failure, the adaptive system is able to quickly reconfigure and track the commanded vertical acceleration, sideslip angle, roll rate, and yaw rate signals, simultaneously. In fact, Figure 4-3a shows that with the adaptation turned on the desired/nominal system tracking behavior has been almost recovered. In addition, Figure 4-3b compares the three virtual control feedback signals, and Figure 4-3c confirms that the level of control activity is reasonable and that no control saturation has occurred during the adaptation process.

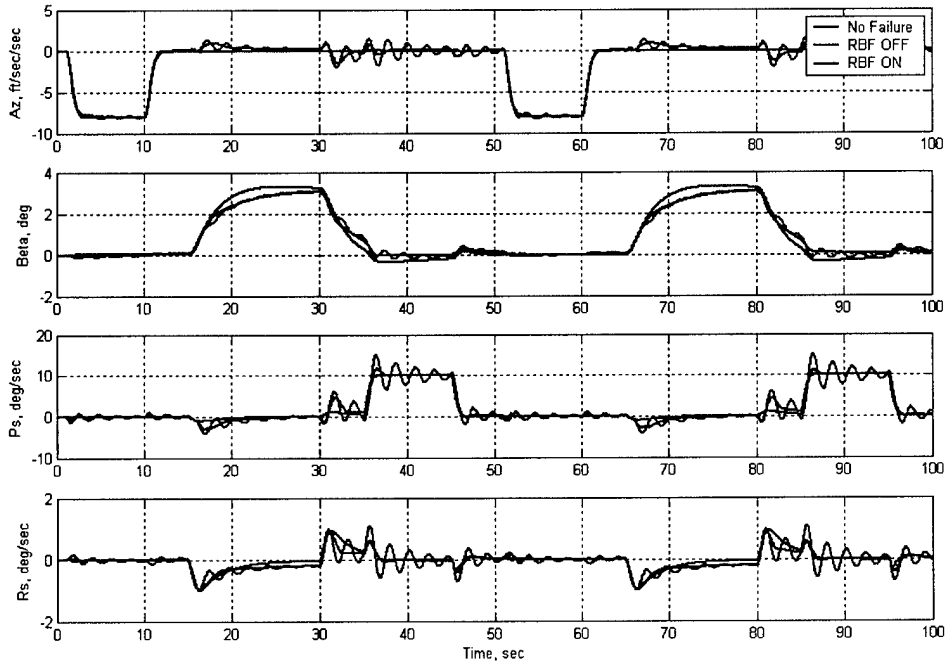


Figure 4-3a. Inner-Loop Adaptation with ROB Elevon Failure: Command Tracking

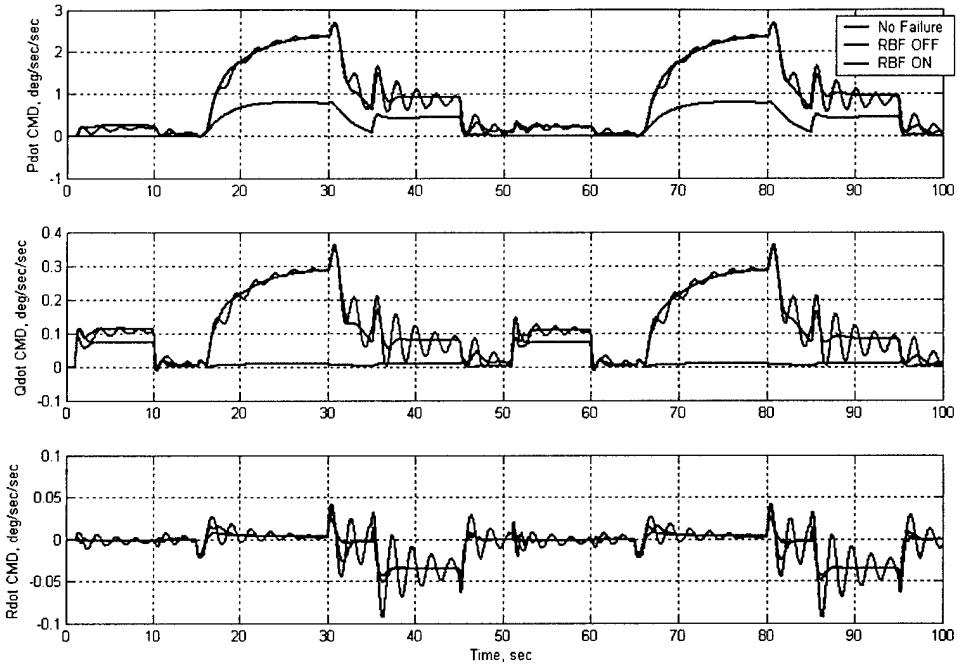


Figure 4-3b. Inner-Loop Adaptation with ROB Elevon Failure: Virtual Controls

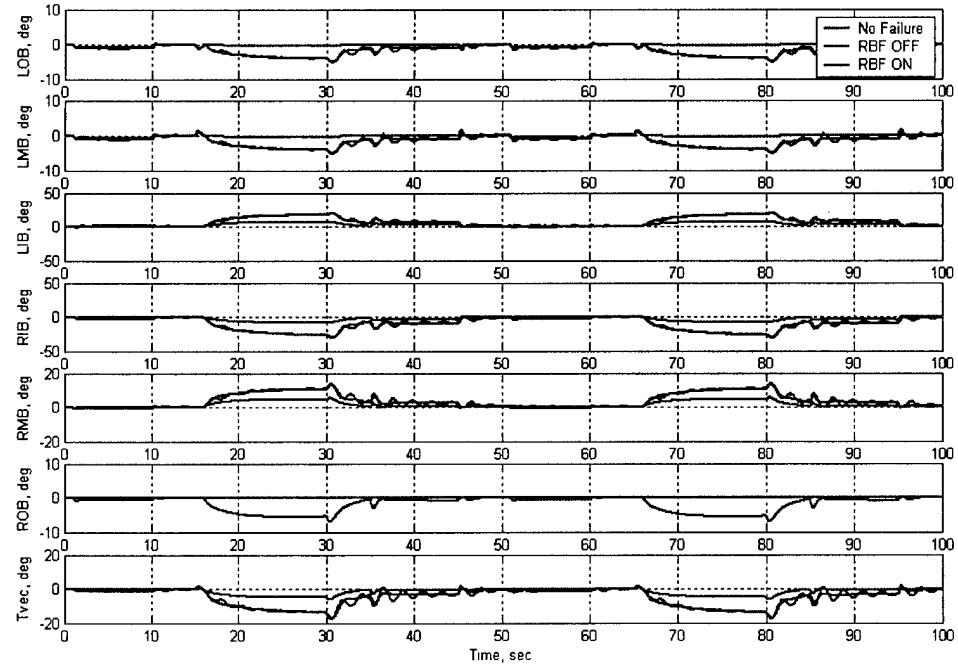


Figure 4-3c. Inner-Loop Adaptation with ROB Elevon Failure: Controls

Figures 4-4a, 4-4b, and 4-4c demonstrate adaptation benefits when in addition to the ROB elevon failure the Pitch Break phenomenon is active throughout the entire maneuver. As seen from the data, the adaptive process removes the unwanted oscillations and quickly "learns" to cancel the unknown Pitch Break nonlinear phenomena. As expected, all of the control signals remain well within their position and rate limits.

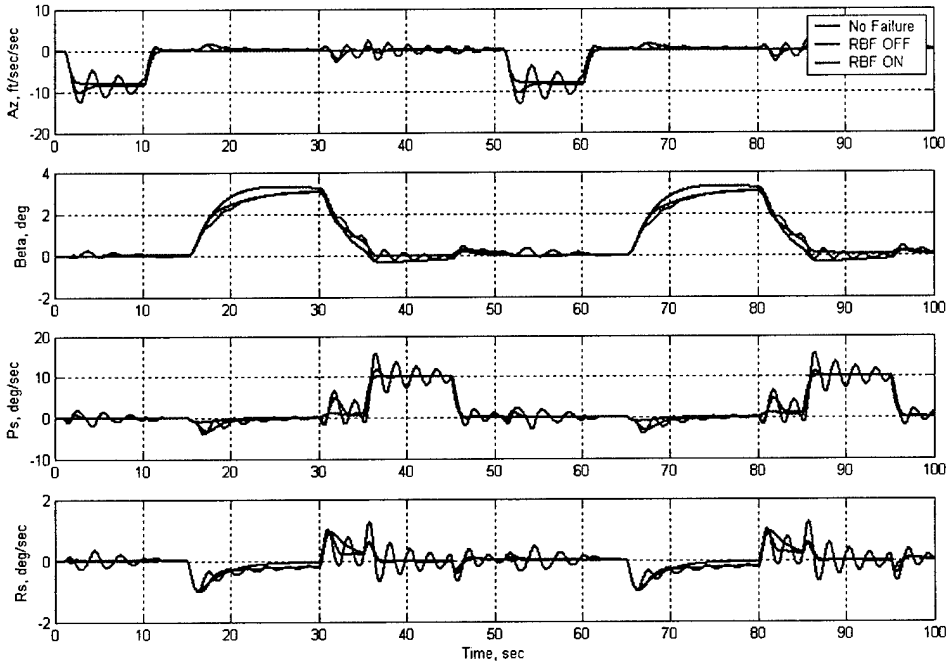


Figure 4-4a. Inner-Loop Adaptation with ROB Elevon Failure and Pitch Break Phenomenon: Command Tracking

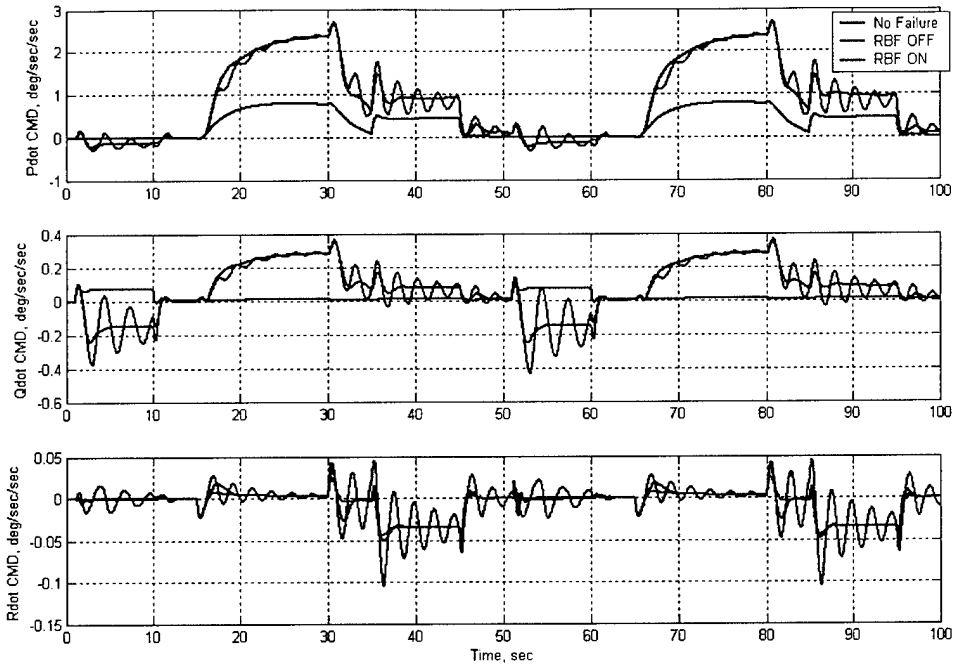


Figure 4-4b. Inner-Loop Adaptation with ROB Elevon Failure and Pitch Break Phenomenon: Virtual Controls

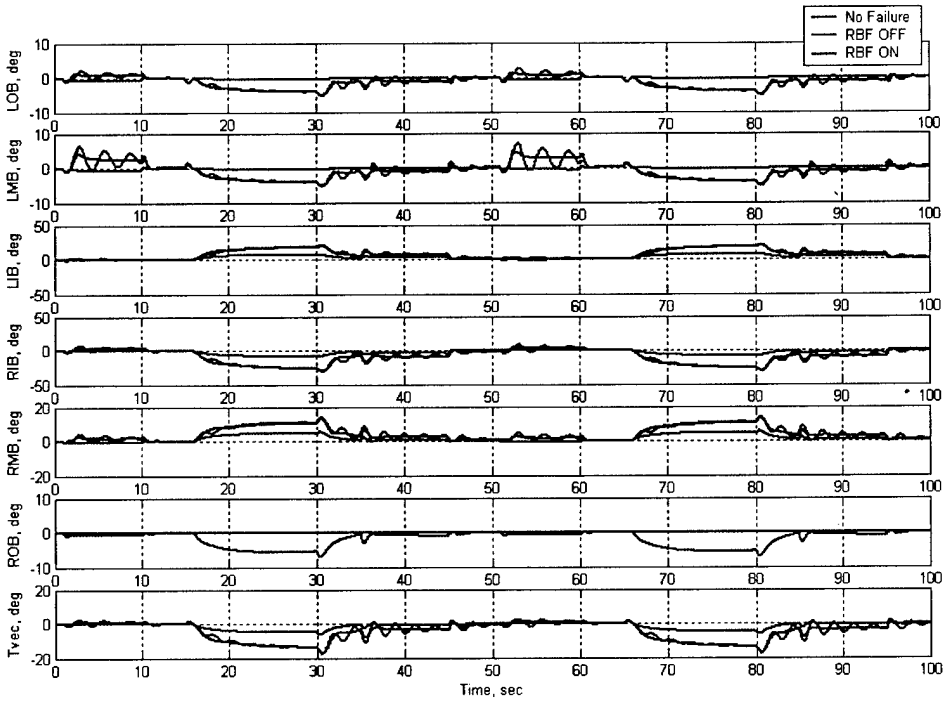


Figure 4-4c. Inner-Loop Adaptation with ROB Elevon Failure and Pitch Break Phenomenon: Controls

Figures 4-5a, 4-5b, and 4-5c demonstrate adaptation benefits when the inner-loop system receives repetitive vertical acceleration commands in the presence of both the ROB elevon failure and the unknown Pitch Break phenomenon. The data in Figure 4-5a show that while the nominal system provides consistent (oscillatory) performance, the adaptive process quickly dampens the oscillations caused by the uncertainties and significantly reduces the corresponding control activity.

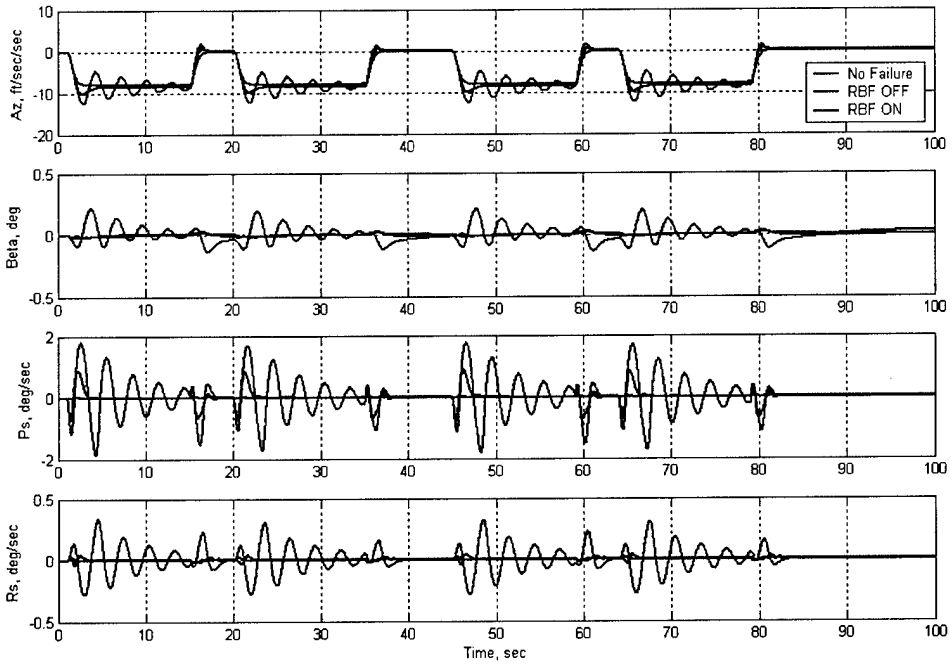


Figure 4-5a. Inner-Loop Adaptation with Vertical Acceleration Commands, ROB Elevon Failure, and Pitch Break Phenomenon: Command Tracking

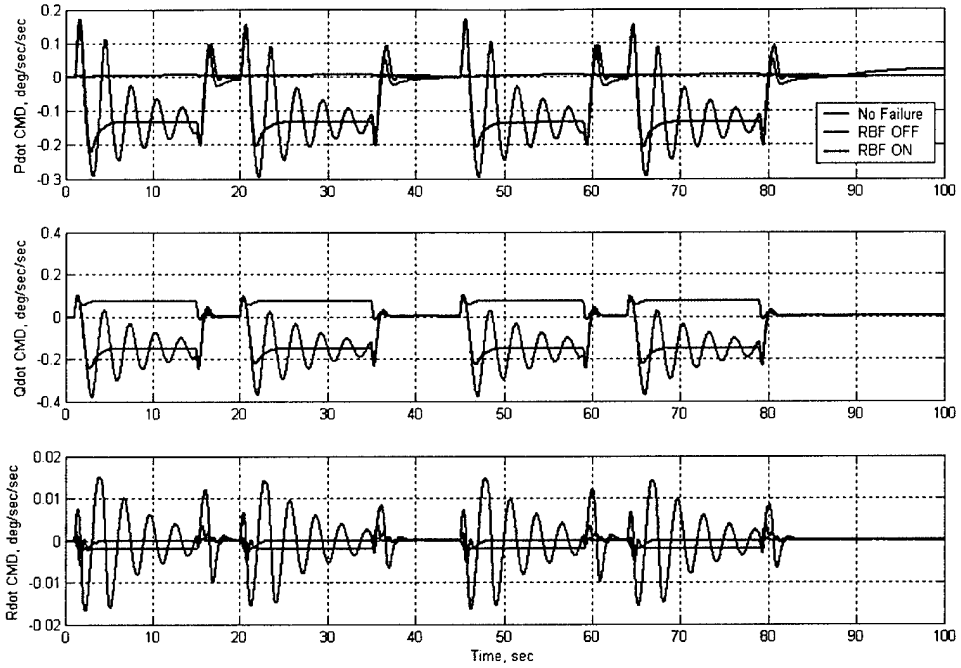


Figure 4-5b. Inner-Loop Adaptation with Vertical Acceleration Commands, ROB Elevon Failure, and Pitch Break Phenomenon: Virtual Controls

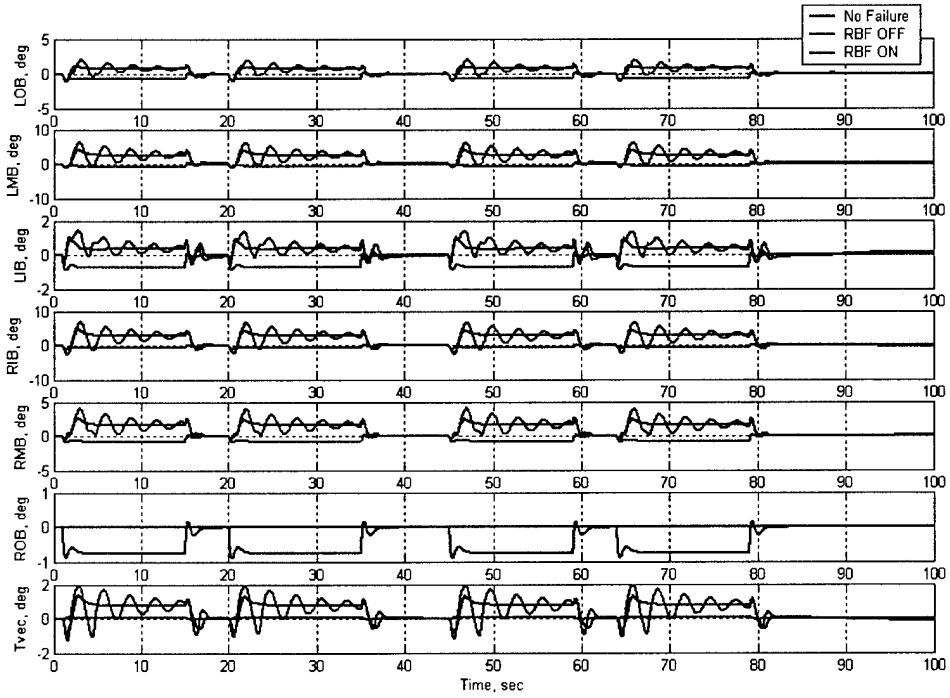


Figure 4-5c. Inner-Loop Adaptation with Vertical Acceleration Commands, ROB Elevon Failure, and Pitch Break Phenomenon: Controls

Adaptive inner-loop system performance was also tested in the UCAV X-45A high fidelity nonlinear 6-DoF flight simulation environment, (MATRIXx). Figure 4-6 compares performance of the three systems: a) Baseline/No Failures (black), b) Baseline/Failures ON (red), and c) Adaptation ON/Failures ON. As seen from the figure, the adaptive system is able to recover its baseline/no failure performance, while accurately tracking the desired inner-loop commands.

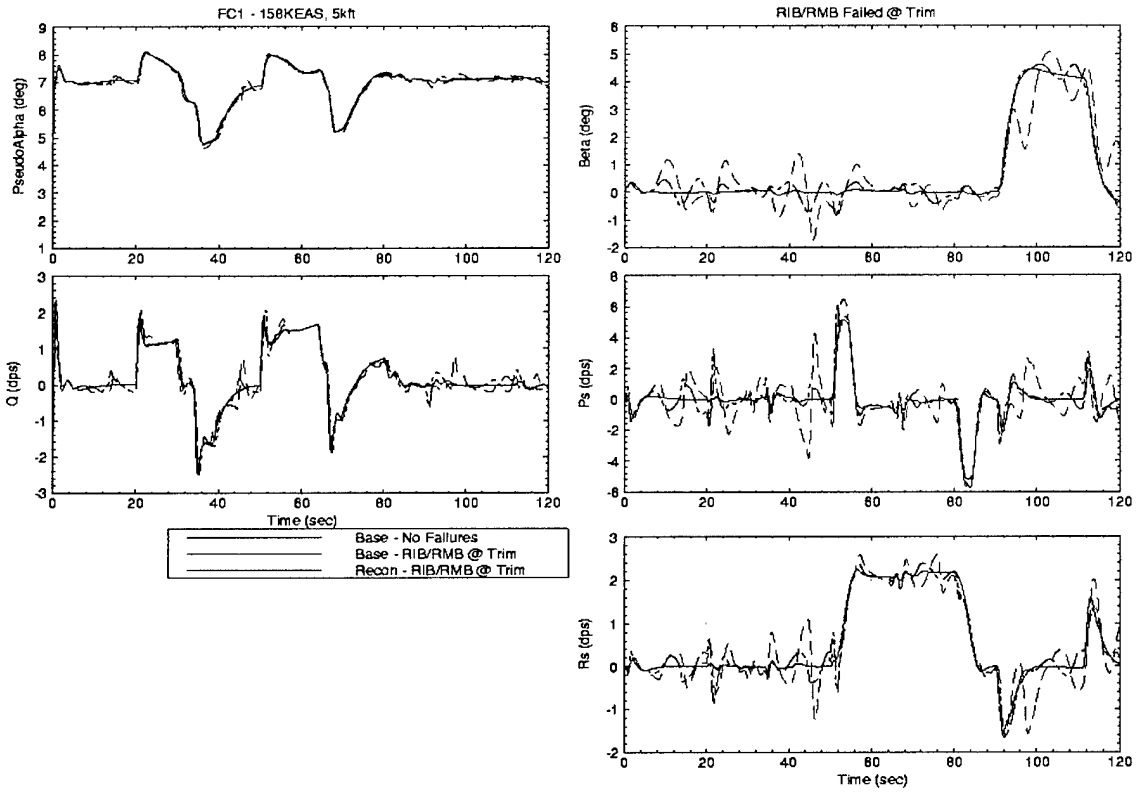


Figure 4-6. Inner-Loop Adaptation with RIB/ROB Elevon Failures @ Trim

The adaptive system performance was evaluated using an extensive matrix of test cases. Linear data for the reference model (4.10) were scheduled throughout the aircraft flight envelope. Required by the adaptive laws (4.34), the unique symmetric positive definite solution P of the corresponding algebraic Lyapunov equation (4.25) was approximated on-line using the iterative procedure outlined in Appendix C.

4.6.2 Internal Model Principle-Based Controller

An IMPNOR control scheme was tested for various inputs and failure conditions. Figure 4-7a shows the simulation results of the inner-loop system with the IMPNOR controller when the right outboard surface (ROB) failure is present. As can be seen, the IMPNOR controller sufficiently recovers the nominal response of the aircraft, by reducing the oscillation. In Figure 4-7a, 4-7b, and 4-7c, the legend IMP is used for IMPNOR.

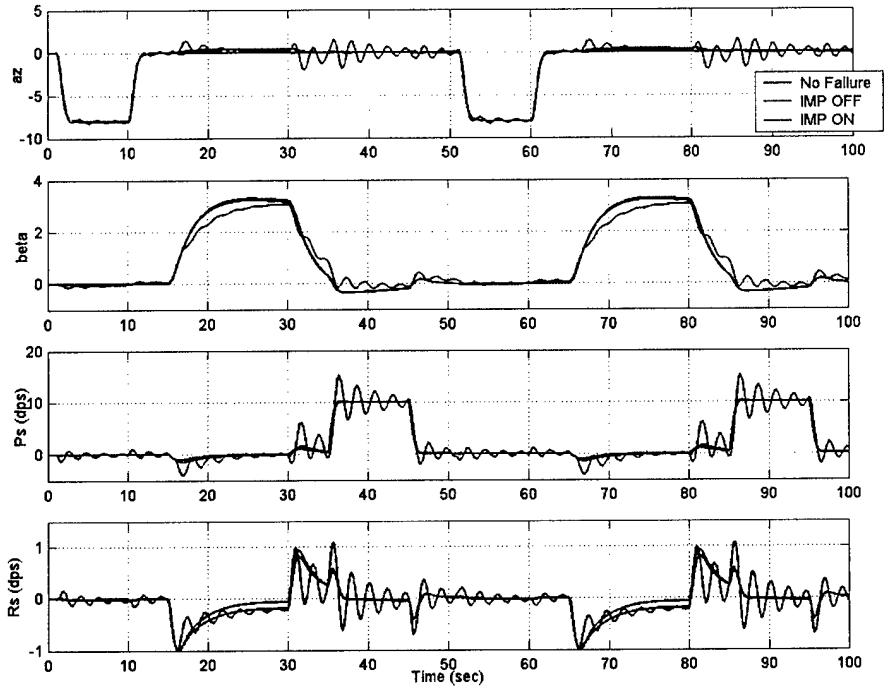


Figure 4-7a. Aircraft Response without Pitch Break Phenomenon with IMPNOR Control

For the body axis angular accelerations, $\dot{P}, \dot{Q}, \dot{R}$ the IMPNOR controller adequately reduces the oscillations that they experience with a ROB failure, as can be seen in Figure 4-7b below. Although the angular accelerations $\dot{P}$ and $\dot{Q}$ still exceeded that of nominal, the aircraft motion is under greater control than that without the IMPNOR controller.

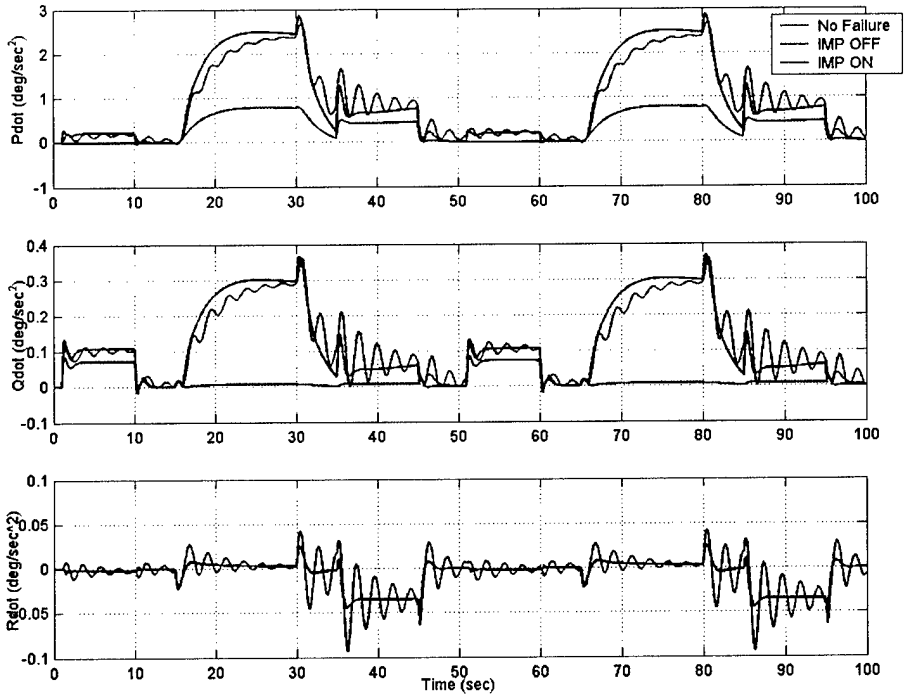


Figure 4-7b. Aircraft Acceleration Response without Pitch Break Phenomenon

Figure 4-7c shows the deflections of the individual surfaces and that the aircraft's surfaces are within actuation position limits and do not suffer from the same oscillations that occur without the IMPNOR controller.

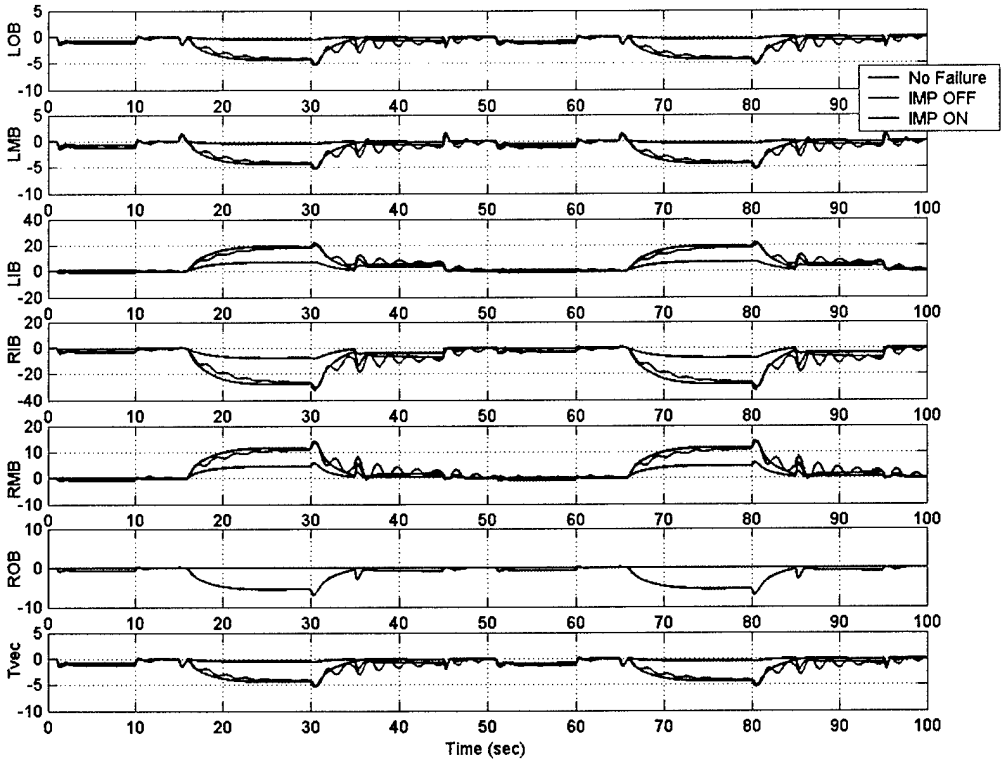


Figure 4-7c. Surface Deflections in Flight Conditions without Pitch Break

Next, the same conditions and inputs were applied to the same aircraft model, only this time with the Pitch Break phenomenon active. From Figures 4-8a, 4-8b, and 4-8c, we see that the IMPNOR controller is able to recover the nominal aircraft response when the right outboard surface failure is present. In fact, the controller is able to dampen out the additional errors induced by the Pitch Break phenomenon in A_z and β better than the baseline controller, giving much better damped responses to all inputs.

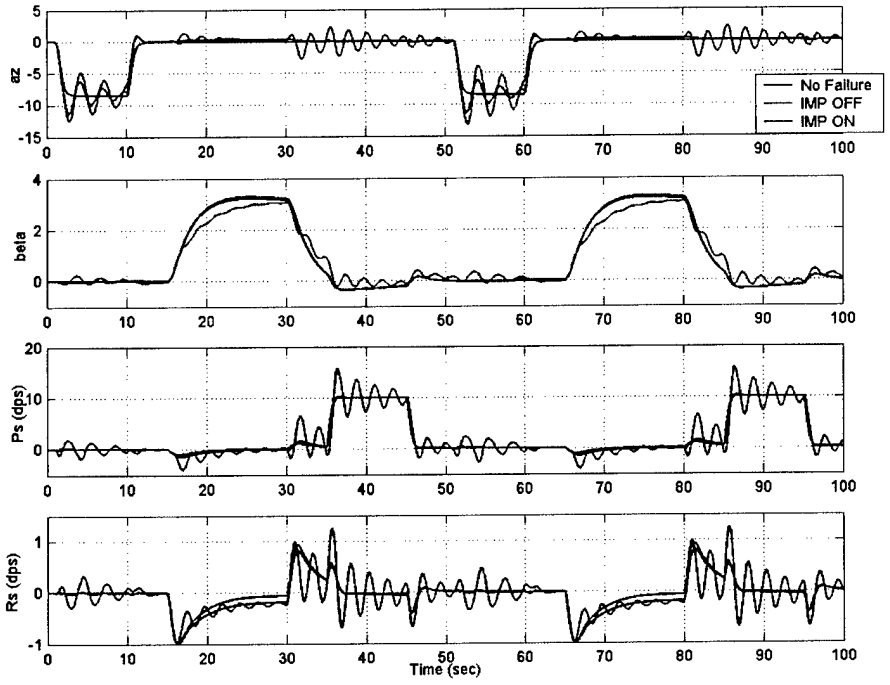


Figure 4-8a. Aircraft Response with Pitch Break Phenomenon

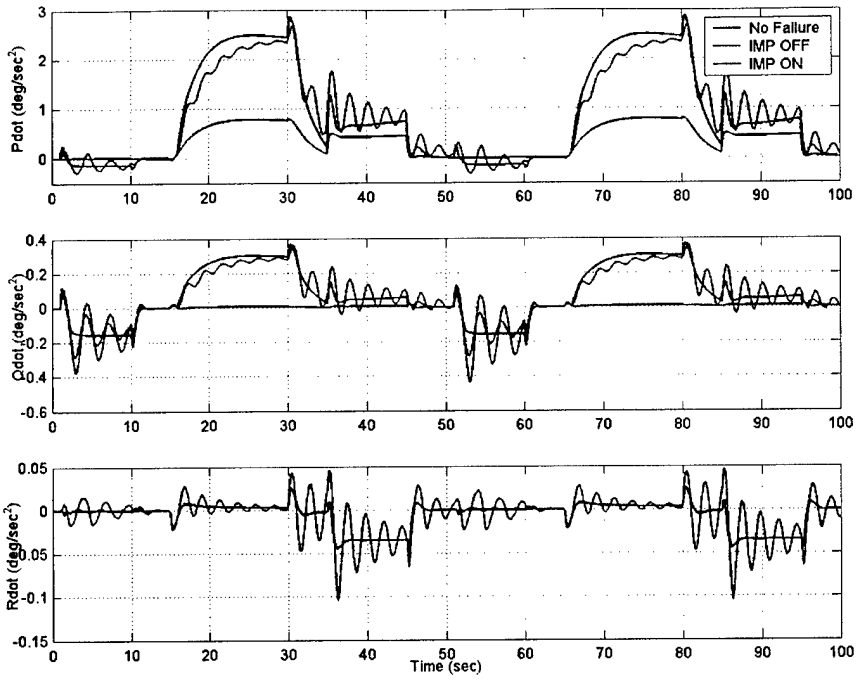


Figure 4-8b. Aircraft Acceleration Response with Pitch Break Phenomenon

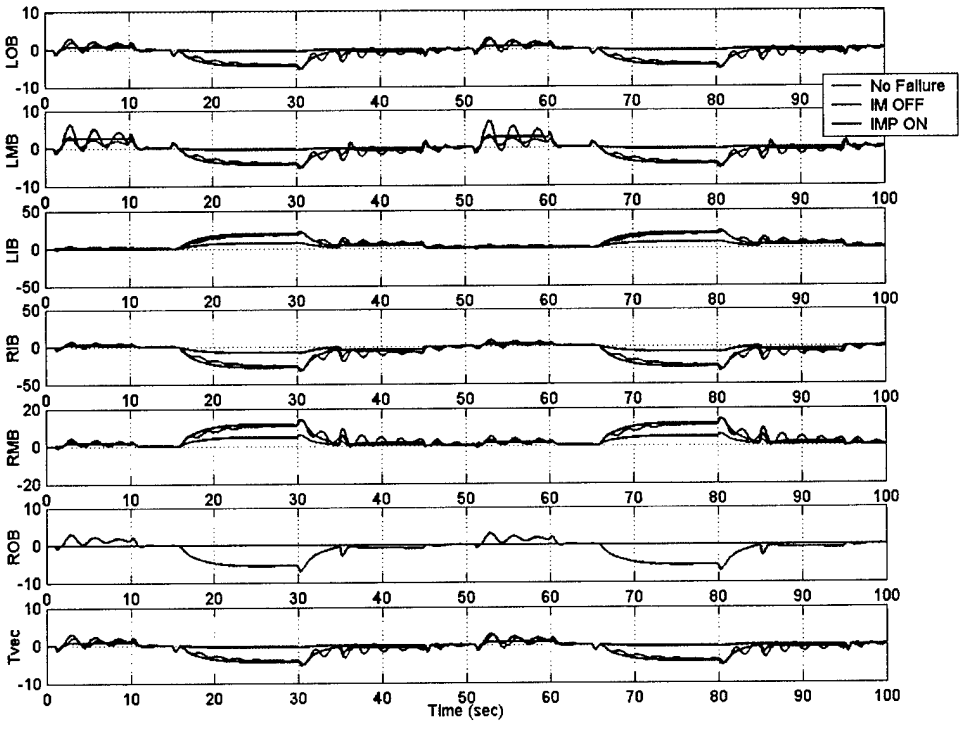


Figure 4-8c. Surface Deflections in Flight Conditions with Pitch Break

4.7 UCAV X-45A Guidance (Outer-Loop) System Augmentation

In this section, the X-45A guidance system is introduced into the adaptive design process. Two guidance models, $n_{long} = 3$ – dimensional longitudinal and $n_{latd} = 2$ – dimensional lateral-directional, are introduced:

$$\begin{cases} \dot{x}_{long} = A_{long}\, x_{long} + B_{long}\, u_{long} \\ y_{long} = C_{long}\, x_{long} + D_{long}\, u_{long} = \ddot{h}^{cmd} \end{cases} \tag{4.37}$$

$$\begin{cases} \dot{x}_{latd} = A_{latd}\, x_{latd} + B_{latd}\, u_{latd} \\ y_{latd} = C_{latd}\, x_{latd} + D_{latd}\, u_{latd} = \phi^{cmd} \end{cases} \tag{4.38}$$

The three dimensional input vector u_{long} to the longitudinal guidance model (4.37) is defined as:

$$u_{long} = \left(h^{cmd} \quad h \quad \dot{h}\right)^T \tag{4.39}$$

where h^{cmd}, h and $\dot{h}$ denote commanded altitude (ft), actual altitude (ft), and aircraft vertical speed (ft/sec). Commanded altitude acceleration $\ddot{h}^{cmd}$ (ft/sec/sec) is computed as the system (4.37) output signal, (that is $y_{long} = \ddot{h}^{cmd}$).

The two dimensional input vector u_{latd} to the lateral-directional guidance model (4.38) is defined as:

$$u_{latd} = \left(\psi^{cmd} \quad \psi\right)^T \tag{4.40}$$

where ψ^{cmd} and ψ denote commanded and actual heading (yaw) angles in degrees. Commanded roll angle ϕ^{cmd} (deg) is computed as the system (4.38) output signal, (that is $y_{latd} = \phi^{cmd}$). Based on the latter, the commanded roll rate p^{cmd} (deg/sec) formed:

$$p^{cmd} = \left(\phi^{cmd} - \phi\right) k_\phi \tag{4.41}$$

In (4.41), positive constant k_ϕ represents a commanded roll gain (1/deg). In addition, the commanded yaw rate r^{cmd} (rad / sec) signal is specified as:

$$r^{cmd} = \frac{g \sin\phi}{V} \tag{4.42}$$

where V denotes the aircraft true airspeed.

Using the two guidance outputs h^{cmd} and ϕ^{cmd}, vertical acceleration command a_z^{cmd} is computed:

$$a_z^{cmd} = -\frac{\ddot{h}^{cmd}}{\cos\left(\phi^{cmd}\right)} \tag{4.43}$$

Combining (4.43), (4.41) and (4.42), the vector of inner-loop commands is formed:

$$u = \left(-\frac{\ddot{h}^{cmd}}{\cos\left(\phi^{cmd}\right)} \quad \beta^{cmd} \quad \left(\phi^{cmd} - \phi\right) k_{\phi} \quad \frac{g \sin\phi}{V} \right)^{T} \tag{4.44}$$

In order to properly account for the 4 extra degrees of freedom $\left(h,\ \dot{h},\ \phi,\ \psi\right)$ used in the guidance model, their approximated dynamics is written as:

$$\begin{cases} \ddot{h} = -a_z \cos\phi \\ \dot{\phi} = p \\ \dot{\psi} = r\cos\phi + q\sin\phi \end{cases} \tag{4.45}$$

Let $B_{long} = \left[b_1^{long} \quad b_2^{long} \quad b_3^{long} \right]$ and $B_{latd} = \left[b_1^{latd} \quad b_2^{latd} \right]$. Augmentation of the (Plant + Inner-Loop Controller) – system (4.8) with: a) extra 4 degrees of freedom (4.45), b) longitudinal guidance (4.37), and c) lateral-directional guidance (4.38), results in the augmented dynamics:

$$\begin{cases} \dot{x} = A x + B_1 \Lambda\left(\delta + K_0\left(x_p\right)\right) + B_2 u \\ \ddot{h} = -\left(\underbrace{\left(C_p \quad 0\right)}_{C} x + D_p \Lambda\left(\delta + K_0\left(x_p\right)\right) \right) \cos\phi \\ \dot{\phi} = p \\ \dot{\psi} = r\cos\phi + q\sin\phi \\ \dot{x}_{long} = A_{long}\, x_{long} + b_1^{long}\, h^{cmd} + b_2^{long}\, h + b_3^{long}\, \dot{h} \\ \dot{x}_{latd} = A_{latd}\, x_{latd} + b_1^{latd}\, \psi^{cmd} + b_2^{latd}\, \psi \end{cases} \tag{4.46}$$

Assuming "small angles" and rewriting (4.46) in a matrix form, we get:

$$\underbrace{\begin{pmatrix} \dot{x} \\ \dot{h} \\ \ddot{h} \\ \dot{\phi} \\ \dot{\psi} \\ \dot{x}_{long} \\ \dot{x}_{latd} \end{pmatrix}}_{\dot{\bar{x}}} = \underbrace{\begin{pmatrix} A & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ -C & 0 & 0 & 0 & 0 & 0 & 0 \\ e_3 & 0 & 0 & 0 & 0 & 0 & 0 \\ e_5 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & b_2^{long} & b_3^{long} & 0 & 0 & A_{long} & 0 \\ 0 & 0 & 0 & 0 & b_2^{latd} & 0 & A_{latd} \end{pmatrix}}_{\bar{A}_0} \underbrace{\begin{pmatrix} x \\ h \\ \dot{h} \\ \phi \\ \psi \\ x_{long} \\ x_{latd} \end{pmatrix}}_{\bar{x}}$$

$$+ \underbrace{\begin{pmatrix} B_1 \\ 0 \\ -D_p \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}}_{\bar{B}_1} \Lambda\left(\delta + K_0\left(x_p\right)\right) + \underbrace{\begin{pmatrix} B_2 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}}_{\bar{B}_2} u + \underbrace{\begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ b_1^{long} & 0 & 0 \\ 0 & 0 & b_1^{latd} \end{pmatrix}}_{\bar{B}_3} \underbrace{\begin{pmatrix} h^{cmd} \\ \beta^{cmd} \\ \psi^{cmd} \end{pmatrix}}_{\bar{u}}$$

or, equivalently:

$$\dot{\bar{x}} = \bar{A}_0 \bar{x} + \bar{B}_1 \Lambda\left(\delta + K_0\left(x_p\right)\right) + \bar{B}_2 u + \bar{B}_3 \bar{u} \tag{4.47}$$

In the formulation of the augmented system matrix $\bar{A}_0$ in (4.47), two unit vectors $(e_3,\ e_5)$ were introduced:

$$\begin{cases} e_3 = (0 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0)^T \\ e_5 = (0 \quad 0 \quad 0 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0 \quad 0)^T \end{cases} \tag{4.48}$$

Additionally, matrix element $\bar{A}_{2,12}$ was set to a nonzero value that represented the aircraft bank angle (φ) influence on the sideslip (β) dynamics.

Let $D_{long} = \begin{bmatrix} d_1^{long} & d_2^{long} & d_3^{long} \end{bmatrix}$, and $D_{latd} = \begin{bmatrix} d_1^{latd} & d_2^{latd} \end{bmatrix}$. Then using again the "small angles" assumption, the vector u of inner-loop commands (4.44) becomes:

$$u = \left(-\ddot{h}^{cmd} \quad \beta^{cmd} \quad \left(\phi^{cmd} - \phi\right) k_\phi \quad \frac{g}{V}\phi \right)^T \tag{4.49}$$

In (4.49), guidance output signals $\ddot{h}^{cmd}$ and ϕ^{cmd} are defined by (4.37) and (4.38), correspondingly.

$$\begin{cases} \ddot{h}^{cmd} = C_{long} \quad x_{long} + d_1^{long} \quad h^{cmd} + d_2^{long} \quad h + d_3^{long} \quad \dot{h} \\ \phi^{cmd} = C_{latd} \quad x_{latd} + d_1^{latd} \quad \psi^{cmd} + d_2^{latd} \quad \psi \end{cases} \tag{4.50}$$

Using relations (4.50) yields:

$$u = \begin{pmatrix} -C_{long}\, x_{long} - d_1^{long}\, h^{cmd} - d_2^{long}\, h - d_3^{long}\, \dot{h} \\ \beta^{cmd} \\ \left(C_{latd}\, x_{latd} + d_1^{latd}\, \psi^{cmd} + d_2^{latd}\, \psi - \phi\right) k_\phi \\ \\ \dfrac{g}{V}\phi \end{pmatrix}$$

or, equivalently, in a matrix form:

$$u = \underbrace{\begin{pmatrix} 0 & -d_2^{long} & -d_3^{long} & 0 & 0 & -C_{long} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -k_\phi & d_2^{latd}\, k_\phi & 0 & C_{latd}\, k_\phi \\ 0 & 0 & 0 & \dfrac{g}{V} & 0 & 0 & 0 \end{pmatrix}}_{\bar{F}_1^T} \underbrace{\begin{pmatrix} x \\ h \\ \dot{h} \\ \phi \\ \psi \\ x_{long} \\ x_{latd} \end{pmatrix}}_{\bar{x}} + \underbrace{\begin{pmatrix} -d_1^{long} & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & d_1^{latd}\, k_\phi \\ 0 & 0 & 0 \end{pmatrix}}_{\bar{F}_2^T} \underbrace{\begin{pmatrix} h^{cmd} \\ \beta^{cmd} \\ \psi^{cmd} \end{pmatrix}}_{\bar{u}} = \bar{F}_1^T\, \bar{x} + \bar{F}_2^T\, \bar{u} \tag{4.51}$$

Finally, substituting (4.51) into (4.47) yields the complete system dynamics:

$$\begin{aligned} \dot{\bar{x}} &= \underbrace{\left(\bar{A}_0 + \bar{B}_2\, \bar{F}_1^T\right)}_{\bar{A}} \bar{x} + \bar{B}_1\, \Lambda\left(\delta + K_0\left(x_p\right)\right) + \underbrace{\left(\bar{B}_3 + \bar{B}_2\, \bar{F}_2^T\right)}_{\bar{B}_4} \bar{u} \\ &= \bar{A}\, \bar{x} + \bar{B}_1\, \Lambda\left(\delta + K_0\left(x_p\right)\right) + \bar{B}_4\, \bar{u} \end{aligned} \tag{4.52}$$

Note that in (4.52) a three dimensional vector of the outer-loop commands $\bar{u} = \left(h^{cmd} \quad \beta^{cmd} \quad \psi^{cmd}\right)^T$ was introduced. From (4.51) it follows that the inner-loop commands u are determined through the outer-loop commands $\bar{u}$ and the system state $\bar{x}$ using the feedforward gain matrix $\bar{F}_2$ and the feedback gain matrix $\bar{F}_1$, correspondingly.

where the symmetric positive definite matrix $\bar{P}$ satisfies algebraic Riccati equation:

$$\bar{A}_{ref}^T \bar{P} + \bar{P} \bar{A}_{ref} - \bar{P} \bar{B}_{ref} \bar{R}^{-1} \bar{B}_{ref}^T \bar{P} + \bar{Q} = 0 \tag{4.57}$$

Using (4.55) and (4.56), the corresponding closed-loop reference model dynamics become asymptotically stable and can be written as:

$$\dot{\bar{x}}_{ref} = \underbrace{\left(\bar{A}_{ref} + \bar{B}_{ref} F_{\bar{x}}^T\right)}_{\bar{A}_{ref}^{cl}} \bar{x}_{ref} + \underbrace{\left(\bar{B}_{ref} F_{\bar{v}}^T\right)}_{\bar{B}_{ref}^{cl}} \bar{v} = \bar{A}_{ref}^{cl} \bar{x}_{ref} + \bar{B}_{ref}^{cl} \bar{v} \tag{4.58}$$

At the same time, feedforward gain $F_{\bar{v}}$ can be chosen to achieve a desired DC gain matrix G_{ref} of the reference model (4.58).

$$\underbrace{-\left(\bar{A}_{ref}^{cl}\right)^{-1} \bar{B}_{ref}}_{\bar{S}_{ref}} F_{\bar{v}}^T = \bar{G}_{ref} \tag{4.59}$$

The unique optimal solution of (4.59) is:

$$F_{\bar{v}}^T = -\underbrace{\left(\bar{S}_{ref}^T \bar{S}_{ref}\right)^{-1} \bar{S}_{ref}^T}_{\bar{S}_{ref}^+} \bar{G}_{ref} = -\bar{S}_{ref}^+ \bar{G}_{ref} \tag{4.60}$$

where $\bar{S}_{ref}^+$ is the pseudo-inverse matrix for the rectangular matrix $\bar{S}_{ref} = \left(\bar{A}_{ref}^{cl}\right)^{-1} \bar{B}_{ref}$.

Note, that the newly derived asymptotically stable reference model (4.58) is similar to the original reference model (4.54). Consequently, without a loss of generality and throughout the paper it is assumed that matrix $\bar{A}_{ref}$ is Hurwitz to begin with, and that the reference model is of the form of (4.54). With that in mind, the outer-inner-loop control objective can now be formulated.

The outer-inner-loop control objective is to design state feedback inner-loop control δ for the complete system (4.52) such that despite the system uncertainties Λ and $K_0\left(x_p\right)$, all closed-loop signals in (4.52) are bounded, and the system state $\bar{x}$ tracks the state $\bar{x}_{ref}$ of the reference model (4.54).

Since the system (4.52) and the reference model dynamics (4.54) are similar to the system (4.8) and its corresponding reference dynamics (4.10), the adaptive control design process is also similar to the design that was introduced in Sections 4.3, 4.4, and 4.5. Repeating the design steps that lead to the inner-loop adaptation (4.34), the new adaptive laws can be written that explicitly account for the guidance loops:

$$\begin{cases} \dot{\hat{k}}_{\bar{x}} = \Gamma_{\bar{x}} \ \mathrm{Proj}\left(\hat{k}_{\bar{x}}, -\bar{x}\bar{e}^T \bar{P} \bar{B}_1\right), \quad \hat{k}_{\bar{x}}(0) = 0_{n\times m} \\ \dot{\hat{k}}_{\bar{u}} = \Gamma_{\bar{u}} \ \mathrm{Proj}\left(\hat{k}_{\bar{u}}, -\bar{u}\bar{e}^T \bar{P} \bar{B}_1\right), \quad \hat{k}_{\bar{u}}(0) = 0_{n\times p} \\ \dot{\hat{\Theta}} = \Gamma_{\Theta} \ \mathrm{Proj}\left(\hat{\Theta}, \Phi\left(x_p\right)\bar{e}^T \bar{P} \bar{B}_1\right), \quad \hat{\Theta}(0) = 0_{N_0\times m} \end{cases} \tag{4.61}$$

In (4.61), $\bar{e} = \bar{x} - \bar{x}_{ref}$ represents the complete system tracking error, $\bar{P}$ is the unique symmetric positive definite solution of the algebraic Lyapunov equation:

$$\bar{A}_{ref}^T \bar{P} + \bar{P} \bar{A}_{ref} = -\bar{Q} \tag{4.62}$$

and $\bar{Q}$ is a symmetric positive definite matrix. In addition, symmetric positive definite matrices $\Gamma_{\bar{x}}$, $\Gamma_{\bar{u}}$, Γ_{Θ} must be chosen to specify desired rates of parameter adaptation.

As it was done in (4.33), the adaptive incremental control is embedded into the nominal inner-loop linear feedback. The total inner-loop control takes the form:

$$\delta = \underbrace{\delta_L\left(x_p, x_c, u\right)}_{\text{nominal}} + \underbrace{\hat{k}_{\bar{x}}^T \bar{x} + \hat{k}_{\bar{u}}^T \bar{u} - \hat{\Theta}^T \Phi\left(x_p\right)}_{\text{adaptive}} \tag{4.63}$$

It is designed to a) change / reconfigure the nominal feedback gains K_x, K_u, b) feed back / feed forward the guidance states / commands, and c) identify / counteract the nonlinear matched uncertainty $K_0\left(x_p\right)$ in the original system (4.1). As a result, bounded tracking of the reference model (4.54) trajectories is achieved in spite of unknown control failures and unmodelled nonlinear dynamical effects. The corresponding closed-inner-outer-loop system block-diagram is shown below.

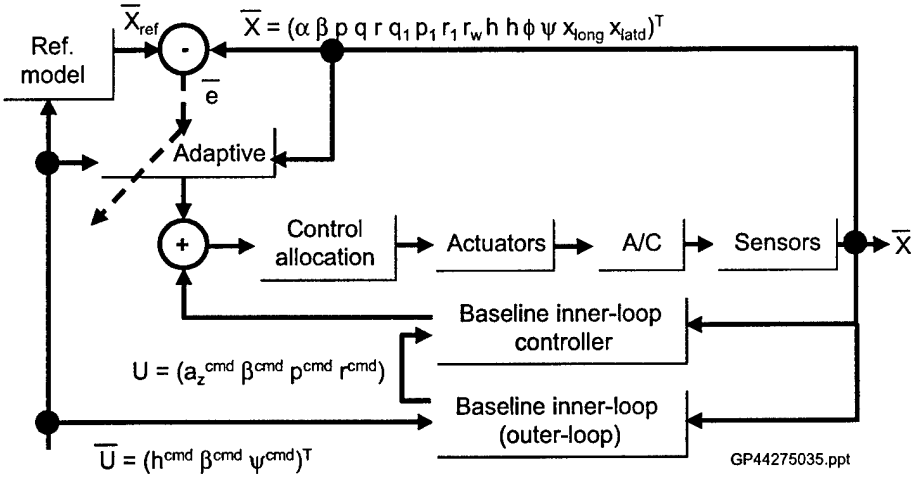


Figure 4-9. Closed-Inner-Outer-Loop System Block Diagram

Note that if the reference model is defined by (4.58) instead of (4.54) then $\bar{P}$ should be taken as the unique solution of the algebraic Riccati equation (4.57). Also, the total inner-loop control input takes the form:

$$\delta = \underbrace{\delta_L\left(x_p, x_c, u\right)}_{\text{nominal}} + \underbrace{\hat{k}_{\bar{x}}^T \bar{x} + \hat{k}_{\bar{u}}^T \underbrace{\left(F_{\bar{x}}^T \bar{x} + F_{\bar{v}}^T \bar{v}\right)}_{\text{guidance augmentation}} - \hat{\Theta}^T \Phi\left(x_p\right)}_{\text{adaptive}} \tag{4.64}$$

with the inner-loop commands vector u defined as in (4.51).

$$u = \bar{F}_1^T \bar{x} + \bar{F}_2^T \bar{u} = \bar{F}_1^T \bar{x} + \bar{F}_2^T \left(F_{\bar{x}}^T \bar{x} + F_{\bar{v}}^T \bar{v} \right)$$
$$= \underbrace{\left(\bar{F}_1^T + \bar{F}_2^T F_{\bar{x}}^T \right)}_{\bar{F}_1^{cl}} \bar{x} + \underbrace{\bar{F}_2^T F_{\bar{v}}^T}_{\bar{F}_2^{cl}} \bar{v} = \bar{F}_1^{cl} \bar{x} + \bar{F}_2^{cl} \bar{v} \qquad (4.65)$$

The corresponding closed-loop system block-diagram is shown in Figure 4-10.

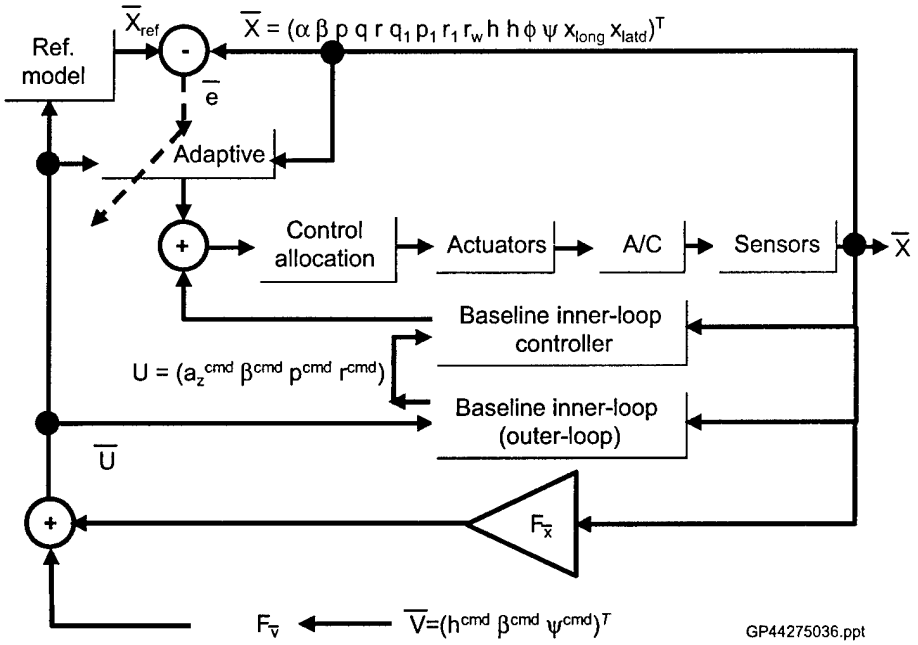


Figure 4-10. Closed-Inner-Outer-Loop System Block Diagram with Guidance Augmentation

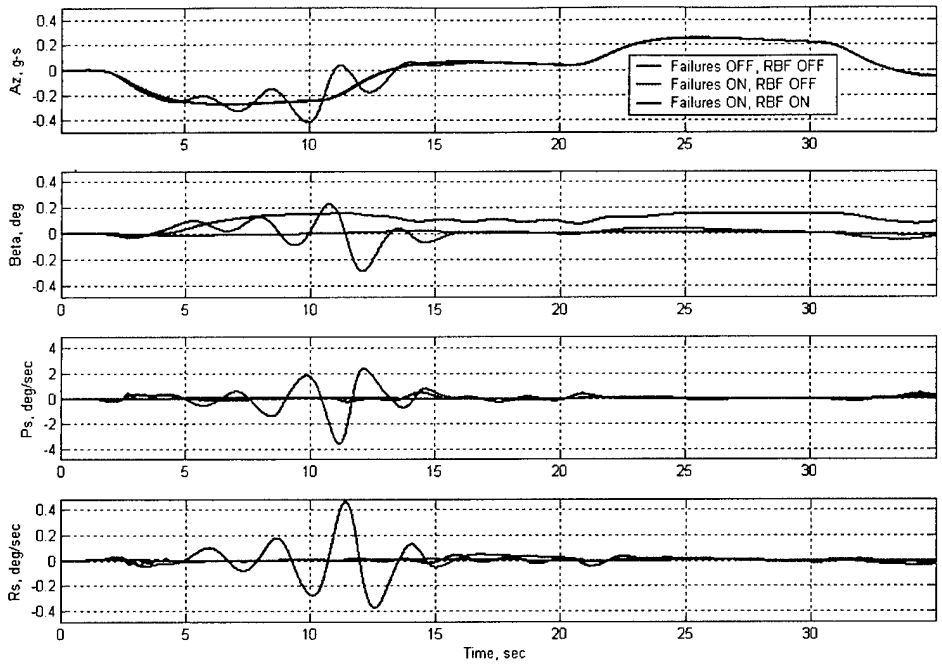


Figure 4-11a. Inner-Outer-Loop Adaptation during Max Rate Climb/Level Off with ROB Elevon Failure and Pitch Break Phenomena

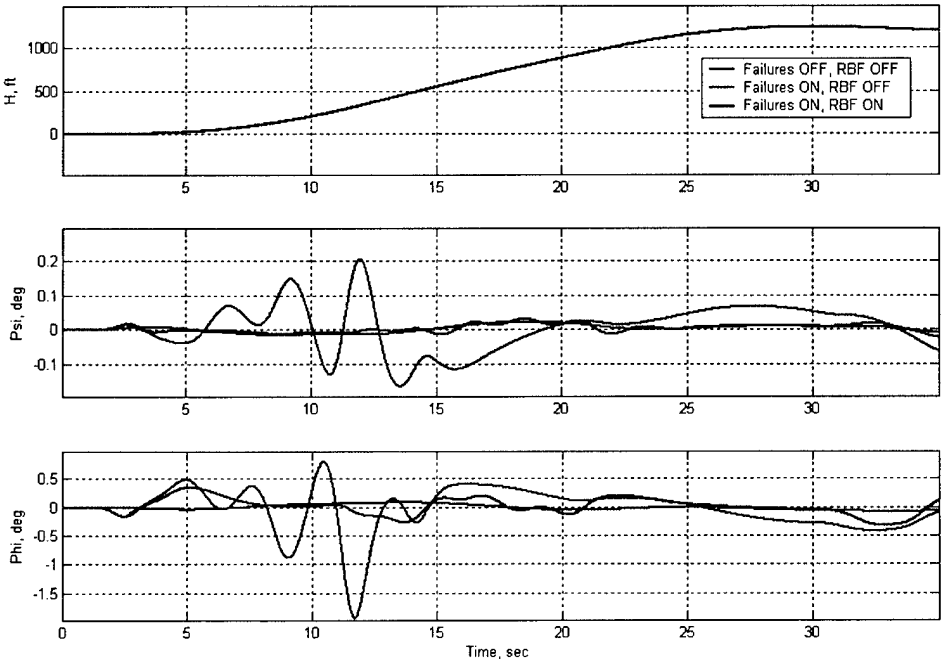


Figure 4-11b. Inner-Outer-Loop Adaptation during Max Rate Climb/Level Off with ROB Elevon Failure and Pitch Break Phenomena

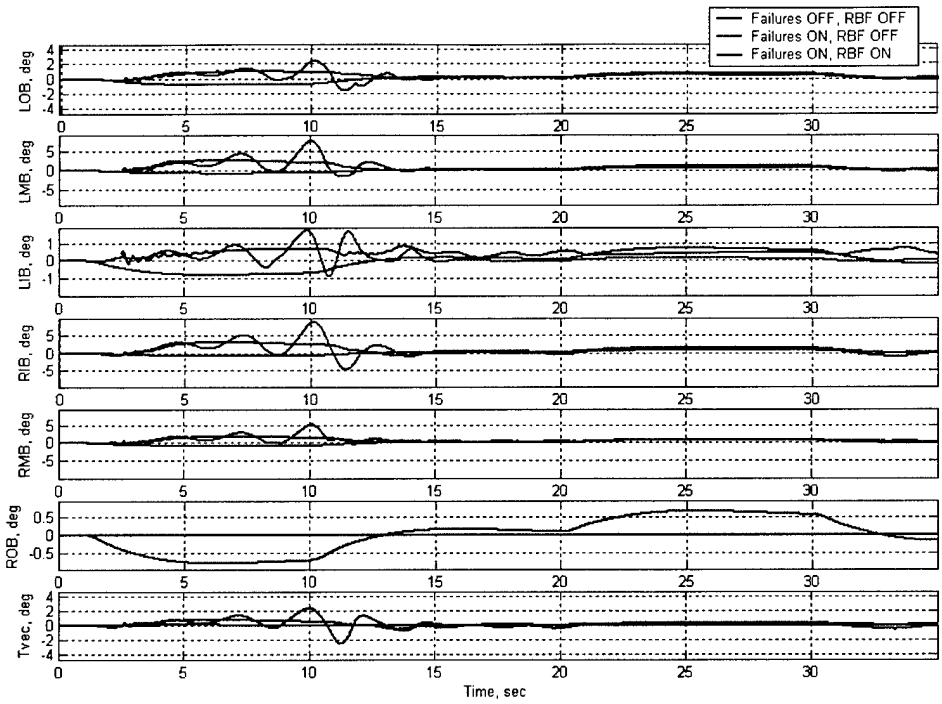


Figure 4-11c. Inner-Outer-Loop Adaptation during Max Rate Climb/Level Off with ROB Elevon Failure and Pitch Break Phenomena: Control Deflections

4.9.2 Internal Model Principle-Based Controller

To evaluate the performance of the IMPNOR controller with the Outer Loop command for a climb and level-off maneuver, simulations were performed and the results are shown in Figure 4-12a, 4-12b, and 4-12c. Note that the Pitch Break phenomenon is active. As can be seen from Figure 4-12a, 4-12b, and 4-12c, unwanted oscillations caused by either the surface failure and/or the Pitch Break phenomenon are effectively cancelled while keeping the surfaces within their actuator position limits.

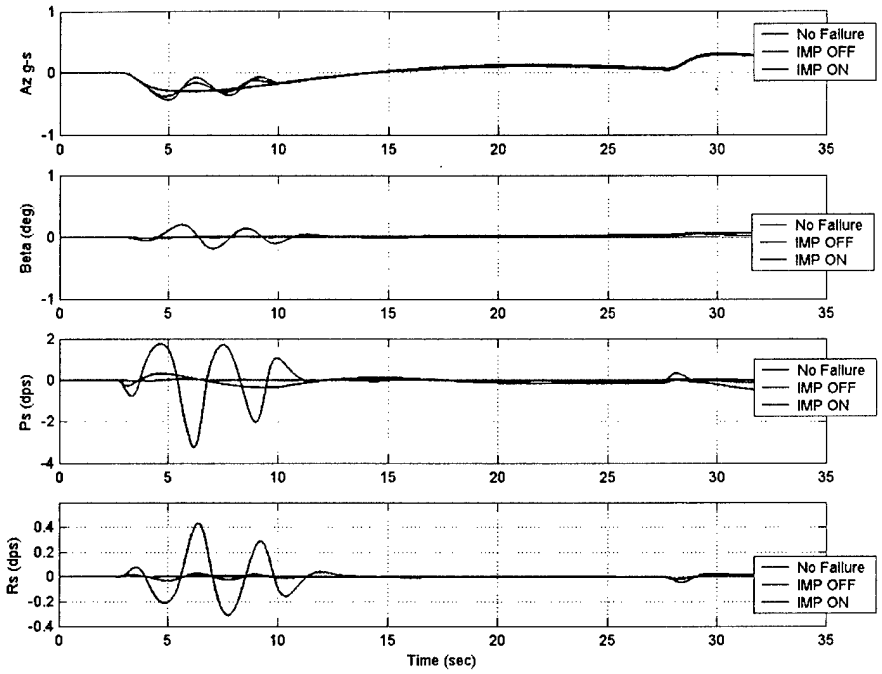


Figure 4-12a. Aircraft Response to Climb/Level-off with Pitch Break Phenomenon and ROB Failure

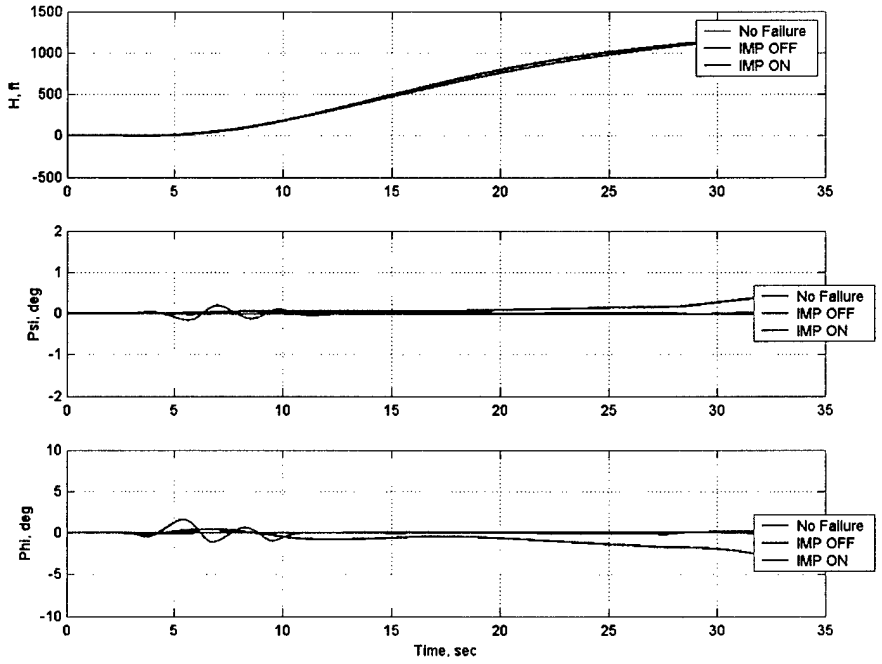


Figure 4-12b. Aircraft Angular Acceleration Response to Climb/Level-off with Pitch Break Phenomenon and ROB Failure

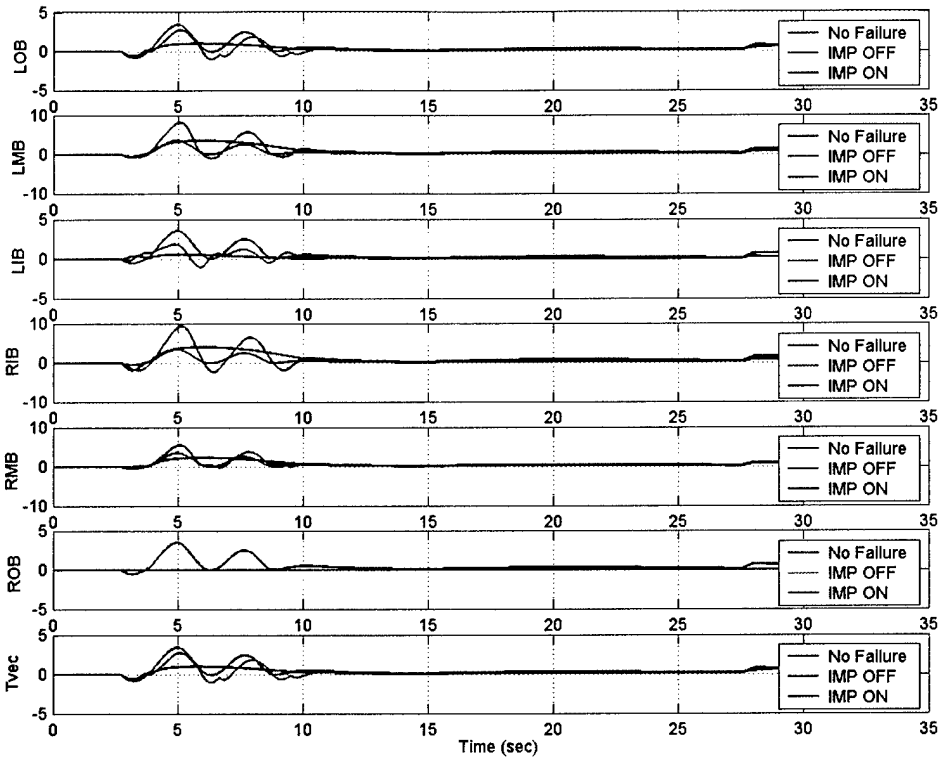


Figure 4-12c. Aircraft Guidance Response to Climb/Level-off with Pitch Break Phenomenon and ROB Failure

4.10 Conclusions

Design, stability analysis and simulation evaluation of a direct adaptive model following control system was presented. The adaptive laws were theoretically justified and derived based on the fundamentals of the Lyapunov Stability Theory. The system was designed to augment an existing nominal robust baseline controller with an adaptive online trained feedforward neural network. The adaptive element provided for reconfiguration in the presence of system uncertainties (e.g., battle damage), control failures, and environmental disturbances. Using UCAV X-45A model and data, a step-by-step adaptive control design process was presented. Radial basis functions (RBF) and single-hidden-layer (SHL) feedforward neural network (NN) architectures in conjunction with the projection operator were employed to derive the adaptive laws. The adaptive element provided bounded tracking and a uniform ultimate boundedness of all the signals in the corresponding closed-loop system. Both the inner-loop design and the outer-inner-loop (explicitly accounting for guidance) adaptations were performed. Benefits of the adaptive system and its performance were evaluated in the UCAV X-45A flight simulation environment.

5.0 ADAPTIVE CONTROL OF ELECTROMAGNETIC ACTUATORS

Actuator nonlinearities are among the key factors limiting both static and dynamic performance of feedback control systems. Harmful effects of backlash in gears are well known. In fact, such nonlinearities were observed and reported in a technical memo [27]. In this memo, it was reported that the quantizer and backlash effects dominated the nonlinear response, and that the response became very nonlinear as the command magnitude (fin deflection angle of a missile) decreased.

Backlash prevents accurate positioning and may lead to chattering and limit-cycle-type instabilities. This increases wear and tear on the gears, which in turn, increases backlash. Various anti-backlash gears have been proposed as a solution to this problem. However, their cost is high, and they introduce extra weight and friction. Attempts to eliminate or reduce backlash by increasing the gain of the feedback loop lead to sustained oscillations which may cause rapid wear of gear trains, valves, and other components. Thus, intelligent but inexpensive approaches to increase accuracy against the nonlinearities in the actuator dynamics are needed in order to improve aircraft and missile performance without additional cost.

G. Tao and P. Kokotovic have done extensive investigation on adaptive control of systems with actuator and sensor nonlinearities [28]. We will follow their methodology (we will call it "Adaptive Inverse Method" in this report) to implement an effective nonlinear control scheme for actuators. We will also apply the Internal Model Principle technique to compare the results with those by the Adaptive Inverse method.

5.1 Inverse Models

The method proposed by Tao and Kokotovic uses an inverse model to cancel the nonlinear effects induced by dead-zone and backlash. That is, if we let u = N(v) represent the nonlinear element in the actuator, where v denotes the input to the actuator and u the output from the actuator, then the compensator for the input nonlinearity N(.) is $v = NI(u_d)$. Here, NI(.) is the desired inverse and u_d is the control input which would achieve the control objective in the absence of N(.). Thus, when the inverse is exact, $u = N(NI(u_d)) = u_d$ achieves the control objectives as if N(.) were absent. However, in reality, NI(.) is seldom exactly known and only an estimate $NI_{cst}(.)$ of the inverse NI(.) can be implemented.

5.1.1 Dead-Zone Inverse

Let b_r (> 0) and b_l (< 0) denote break-points, and m_r and m_l, represent the slopes of the linear portion of the dead-zone characteristics as shown in Figure 5-1. For a linear parametrization of a dead-zone inverse, we use the estimates $\hat{m_r b_r}$, $\hat{m_r}$, $\hat{m_l b_l}$, $\hat{m_l}$ of $m_r b_r$, m_r, $m_l b_l$, m_l, respectively. The estimates of b_r, b_l are then obtained from

$$\hat{b_r} = \frac{\hat{m_r b_r}}{\hat{m_r}}$$

$$\hat{b_l} = \frac{\hat{m_l b_l}}{\hat{m_l}} \qquad (5.1)$$

Then the inverse model for the dead-zone characteristic is described [24] by

$$
\begin{aligned}
v(t) = DI_{cst}(u_d(t)) &= \frac{u_d(t) + \hat{m}_r \hat{b}_r}{\hat{m}_r} && \text{if } u_d(t) > 0 \\
&= 0 && \text{if } u_d(t) = 0 \\
&= \frac{u_d(t) + \hat{m}_l \hat{b}_l}{\hat{m}_l} && \text{if } u_d(t) < 0
\end{aligned} \tag{5.2}
$$

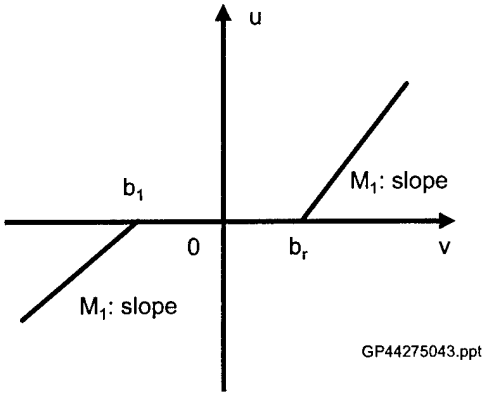


Figure 5-1. Dead-zone

5.1.2 Backlash Inverse

One of the damaging effects of backlash on system performance is the delay corresponding to the time needed to traverse an inner segment. Another undesirable effect of backlash is the information loss occurring on an inner segment when the output u(t) remains constant while the input v(t) continues to change as shown in Figure 5-2.

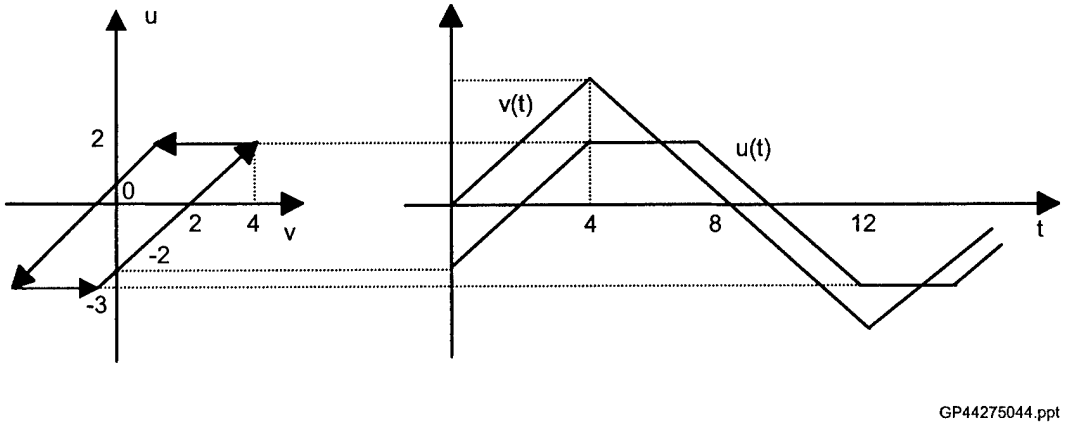


Figure 5-2. Backlash Response to a Sawtooth Input

These undesirable effects are eliminated with the backlash inverse v(t) = BI(u_d(t)) where v(t) is determined by

$$\dot{v}(t) = \frac{1}{m}\dot{u}_d(t) \text{ if } \dot{u}_d(t) > 0, v(t) = \frac{u_d(t)}{m} + c_r, \text{ or}$$

$$\text{if } \dot{u}_d(t) < 0, v(t) = \frac{u_d(t)}{m} + c_l$$

$$= 0 \qquad \text{if } \dot{u}(t) = 0$$

$$= g(t,t) \qquad \text{if } \dot{u}(t) > 0, v(t) = \frac{u_d(t)}{m} + c_l$$

$$= -g(t,t) \qquad \text{if } \dot{u}(t) < 0, v(t) = \frac{u_d(t)}{m} + c_r. \tag{5.3}$$

In this definition, the inverse of a horizontal segment of the backlash characteristic is a vertical jump defined as the time integral of the impulse function

$$g(\tau,t) = \delta(\tau - t)(c_r - c_l) \tag{5.4}$$

where δ(t) is the Dirac δ-function. Thus an upward jump in the backlash inverse is

$$v(t^+) = v(t^-) + \int_{t^-}^{t^+} g(\tau,t)d\tau = \frac{u_d(t^-)}{m} + c_r. \tag{5.5}$$

The effect of this jump in the backlash inverse model will be to eliminate the delay caused by a segment in the backlash characteristic. In a similar manner, the use of the backlash inverse model given by (5.3-5.5) restores the information that would have been lost without it. When the backlash parameters m, c_l, c_r are unknown, we will replace m, c_l, and c_r with their estimates $\hat{m}$, $\hat{c}_l$, $\hat{c}_r$ in equations (5.3-5.5) to design a backlash inverse estimate.

5.2 Adaptive Inverse Control

The objective is to control a system with a nonlinearity N(.) at the input of a linear time-invariant part G(s), which is described by

$$y(t) = G(t)\,[u](t),\ u(t) = N(v(t)), \tag{5.6}$$

where v(t) is the control input and y(t) is the measured output. The linear part is $G(s) = k\frac{Z(s)}{P(s)}$, where k is a constant gain and Z(s), P(s) are monic polynomials in the Laplace transform variable s. Our goal is to develop a method to design a feedback control v(t) using the measured output y(t) to achieve global stabilization and close tracking of a reference signal y_m(t) by y(t).

The control strategy proposed by Tao and Kokotovic uses a linear model reference controller structure plus an adaptive nonlinear inverse model to be placed at the input of the nonlinear element. To employ a model reference approach, they characterize the reference signal $y_m(t)$ as

$$y_m(t) = W_m [r] (t) \tag{5.7}$$

where W_m is stable and the command input r(t) is bounded and piecewise differentiable and impose the following assumption:

A1) Z(s) is a stable polynomial,

A2) The degree n of P(s) is known,

A3) The relative degree $n^* = n - m$ of G(s) is known,

A4) The sign of the gain k is known,

A5) $W_m(s) = P_m^{-1}(s)$ for a stable polynomial $P_m(s)$ of degree equal to the relative degree n*.

With these assumptions together with the assumption that excludes the inappropriate range for parameter, they established the controller structure shown in Figure 5-3.

If $\dot{y}(t)$ and $\dot{r}(t)$ are both available and, $\dot{r}(t)$ is bounded, then the following controller structure is suitable:

$$u_d(t) = \theta_1^{*T} \omega_1(t) + \theta_2^{*T} \omega_2(t) + \theta_{20}^{*} y(t) + \theta_3^{*} r(t) \tag{5.8}$$

where θ_1^* and θ_2^* are vector parameters; and θ_{20}^*, θ_3^* are scalar parameters.

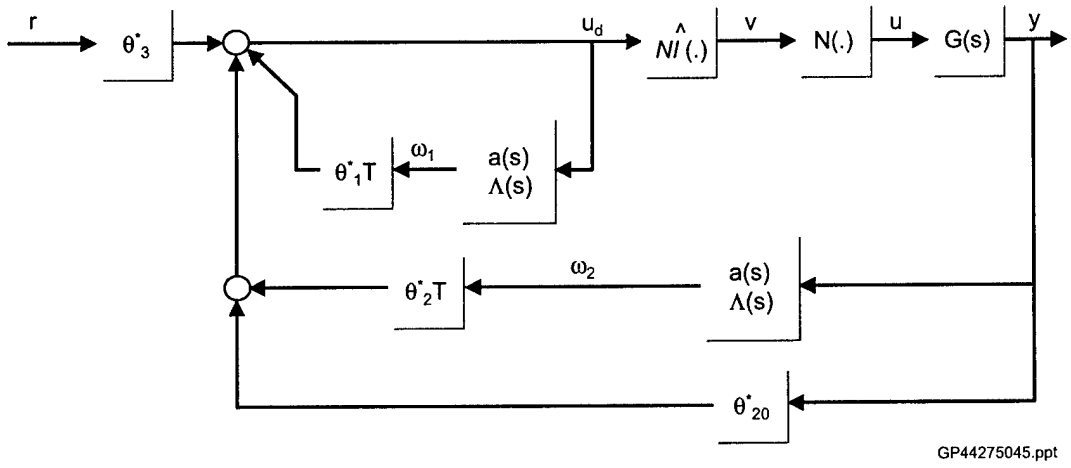


Figure 5-3. Controller Structure for G(s) Known

In the filters

$$\omega_1(t) = \frac{a(s)}{\Lambda(s)}[u_d](t), \qquad \omega_2(t) = \frac{a(s)}{\Lambda(s)}[y](t), \tag{5.9}$$

the vector a(s) is

$$a(s) = (1, s, \ldots, s^{n-2})^T$$

and $\Lambda(s)$ is a stable polynomial of degree n-1.

The parameters

$$\theta_1^* \in R^{n-1}, \theta_2^* \in R^{n-1}, \theta_{20}^* \in R, \theta_3^* \in R$$

are chosen to satisfy the matching equation

$$\theta_1^{*T} a(s)P(s) + (\theta_2^{*T} a(s) + \theta_{20}^* \Lambda(s))kZ(s) = \Lambda(s)(P(s) - \theta_3^* kZ(s)P_m(s)) \tag{5.10}$$

With this controller structure a dead-zone inverse is directly implementable, while the backlash inverse requires $\dot{u}_d(t)$, which can be obtained by

$$\dot{u}_d(t) = \theta_1^* \frac{sa(s)}{\Lambda(s)}[u_d](t) + \theta_2^{*T} \frac{sa(s)}{\Lambda(s)}[y](t) + \theta_{20}^* \dot{y}(t) + \theta_3^* \dot{r}(t). \tag{5.11}$$

An alternative controller structure is also developed for the case when $\dot{y}(t)$ is not available, but $\dot{r}(t)$ is measured and bounded (see [24]). If $\dot{y}(t)$ or $\dot{r}(t)$ is not available, then we can employ a simple approximation of $\dot{u}_d(t)$ such as

$$\dot{u}_d(t) \approx \frac{s}{\varpi + 1}[u_d](t) \tag{5.12}$$

where $\tau > 0$ is a small constant and $\dot{y}(t)$ and $\dot{r}(t)$ as

$$\dot{y}(t) \approx \frac{s}{\varpi + 1}[y](t), \qquad \dot{r}(t) \approx \frac{s}{\varpi + 1}[r](t). \tag{5.13}$$

Based on the tracking error equation derived by Tao and Kokotovic,

$$e(t) = W(s)[(\theta_N - \theta_N^*)^T \omega_N](t) + d(t) \tag{5.14}$$

where $d(t) = W(s)[d_N](t)$, the following update law can be adopted:

$$\dot{\theta}_N(t) = -\frac{\Gamma_N \varsigma_N(t)\varepsilon_N(t)}{m^2{}_N(t)} + f_N(t) \tag{5.15}$$

where $f_N(t)$ is a modification term,

$$\begin{aligned}
&\varepsilon_N(t) = e(t) + \xi_N(t) \\
&\xi_N(t) = \theta_N{}^T(t)\varsigma_N(t) - W(s)[\theta_N{}^T\omega_N](t) \\
&\varsigma_N(t) = W(s)[\omega_N](t) \\
&m_N(t) = \sqrt{1 + \varsigma_N{}^T(t)\varsigma_N(t) + \xi_N{}^2(t)} \\
&\Gamma_N = diag\{\gamma_1,...,\gamma_{n_N}\}, \gamma_i > 0, i = 1,...,n_N \\
&\theta_N = \theta_d = (\hat{m}_r, \hat{m}_r b_r, \hat{m}_l, \hat{m}_l b_l)^T \; or \; \theta_b = (\hat{mc}_r, \hat{m}, \hat{mc}_l)^T
\end{aligned} \tag{5.16}$$

and n_N is the dimension of θ_N.

To Evaluate the performance of the Adaptive Inverse Control scheme, simulation was performed using the Miniaturized Munition Technology actuator model shown in Figure 5-4. The model contains nonlinearities such as deadzone, Coulomb friction in the motor, Coulomb friction in the load, and Quantization errors. The simulation results showed a potential that the Adaptive Inverse Control technique can provide an improved performance in controlling the actuator dynamics compared to a controller with a detuned fixed inverse or no inverse. However, the simulation results also showed the following issues that need to be addressed for practical implementation of the method:

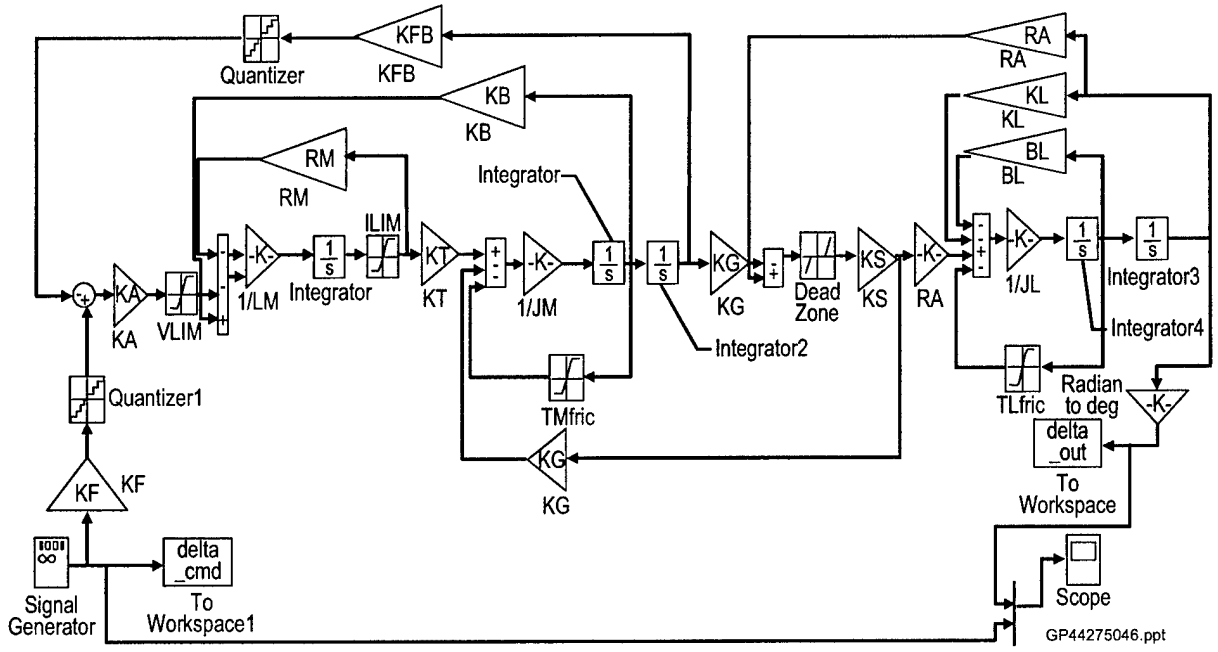


Figure 5-4. Matlab MMT Model

1. Control chatter – "softening" the inverse could reduce the chatter at the cost of increased inversion errors.
2. Adaptation gain –lack of this logic made the adaptation process extremely slow.
3. Robust adaptation – need of robust adaptation logic that would allow larger initial parameter errors at the expense of increased tracking error bounds.

We consider an alternative method that alleviates these potential problems.

5.3 Application of Internal Model Principle Output Regulation for Actuator Control

In this section the internal model principle is applied to the actuator control problem where there exists a dead zone that limits the control authority needed to achieve some desired task. Recall that graphically, the dead zone can be represented by a linear piece-wise continuous input-output map.

$$DZ(v)=\begin{cases} s_2(v-a_2)+h & v \geq a_2 \\ h & -a_1 < v < a_2 \\ s_1(v+a_1)+h & v \leq -a_1 \end{cases}$$

where s_1 and s_2 are the slopes of the input-output graph to the left and to the right of the dead zone area, respectively, $-a_1$ and a_2 are the dead zone left and right boundaries, respectively, and h is the vertical offset. The dead zone problem has been addressed by Tao and Kokotovic [28] using adaptive control methods to realize a dead zone inverse, denoted by $DZ^{-1}(u)$, to cancel the effect of the dead zone. When the parameters of the dead zone are known, then the dead zone inverse can be constructed easily. On the other hand, when the dead zone parameters are uncertain, the dead zone inverse can be reconstructed by an adaptive method or, in this case, via an internal model. The mathematical representation of the inverse dead zone is

$$DZ^{-1}(u)=\begin{cases} \frac{u-h}{s_2}+a_2 & u > h \\ 0 & u = h \\ \frac{u-h}{s_1}-a_1 & u < h \end{cases}$$

It can be shown that $DZ(DZ^{-1}(u))=u$ and $DZ^{-1}(DZ(v))=v$. The adaptive method attempts to estimate the unknown parameters of the inverse dead zone, whereas the internal model method seeks to estimate the required output of the inverse dead zone.

To understand the method first consider the simple system

$$\dot{x} = bx + DZ(v)$$

where -1≤ b≤ 1 and the dead zone parameters are uncertain but are known to lie on some range. Now assume that the goal is to track the output x_{rcf} of the reference system

$$\begin{aligned} \dot{w}_1 &= w_2 \\ \dot{w}_2 &= -w_1 \\ x_{ref} &= w_1 \end{aligned}$$

If b is known and the input to the plant is

$$v = DZ^{-1}(k_0 e + \dot{x}_{ref} - bx_{ref})$$

then the problem is solved for some $k_0=k_0(b,R(Dzp))$ where R(Dzp) represents the range of the dead zone parameters. The rate of convergence is determined by the choice of k_0. Since the parameters are not known at any given time, this input cannot be constructed directly. Therefore, the desired input at any given time is

$$v = DZ^{-1}(u) = \begin{cases} \dfrac{u-h}{s_2} + a_2 & u > h \\ 0 & u = h \\ \dfrac{u-h}{s_1} - a_1 & u < h \end{cases}$$

where

$$u = k_0 e + \dot{x}_{ref} - bx_{ref}$$

Given the ranges of the parameters, it is possible to choose k_0 for robust stability, however, the gains on the reference based-terms must be specified exactly. Since x_{rcf} is generated by the dynamic reference model, an immersion is sought to bypass this requirement. The reference dynamics with output u

$$\begin{aligned} \dot{w}_1 &= w_2 \\ \dot{w}_2 &= -w_1 \\ u &= a_1 w_1 + a_2 w_2 + a_3 \end{aligned} \quad \text{is immersed into} \quad \begin{aligned} \dot{\xi} &= \phi(\xi) \\ u &= \gamma(\xi) \end{aligned}$$

The theory of system immersion is outlined in [28] and will not be repeated here. The important point is system immersion allows for the realization of a controller whose output is a copy of the steady-state input needed to regulate the stabilized system despite bounded uncertainty in the plant parameters.

The controller that is used is

$$\begin{aligned} \dot{\xi}_1 &= \xi_2 + n_1 e \\ \dot{\xi}_2 &= \xi_3 + n_2 \dot{e} \\ \dot{\xi}_3 &= -\xi_2 + n_3 e \\ \dot{\xi}_0 &= K\xi_0 + Le \end{aligned}$$

where the parameters $\{n_1,n_2,n_3,K,L,M,\Gamma_1\}$ are chosen to robustly stabilize the interconnection, yielding satisfactory tracking for small and large inputs. The small inputs are more affected by the dead zone than the larger inputs as will be shown.

$$\begin{aligned}
\dot{x} &= bx + DZ(v) \\
\dot{\xi}_1 &= \xi_2 + n_1 e \\
\dot{\xi}_2 &= \xi_3 + n_2 e \\
\dot{\xi}_3 &= -\xi_2 + n_3 e \\
\dot{\xi}_0 &= K\xi_0 + Le \\
v &= \Gamma_1\xi_1 + M\xi_0 + k_0 e
\end{aligned}
\qquad
\begin{aligned}
\dot{w}_1 &= w_2 \\
\dot{w}_2 &= -w_1 \\
e &= x - w_1
\end{aligned}$$

$$DZ(v) = \begin{cases} s_2(v-a_2)+h & v \geq a_2 \\ h & -a_1 < v < a_2 \\ s_1(v+a_1)+h & v \leq -a_1 \end{cases}$$

where $-1 \leq b \leq 1$, the dead zone parameters satisfy the constraints $s_1, s_2 \in [1,5]$, $h \in [-1,1]$, $a_1, a_2 \in [0.1,1]$. The single state of the plant was rate limited (±100 units) and position-limited (±20 units).

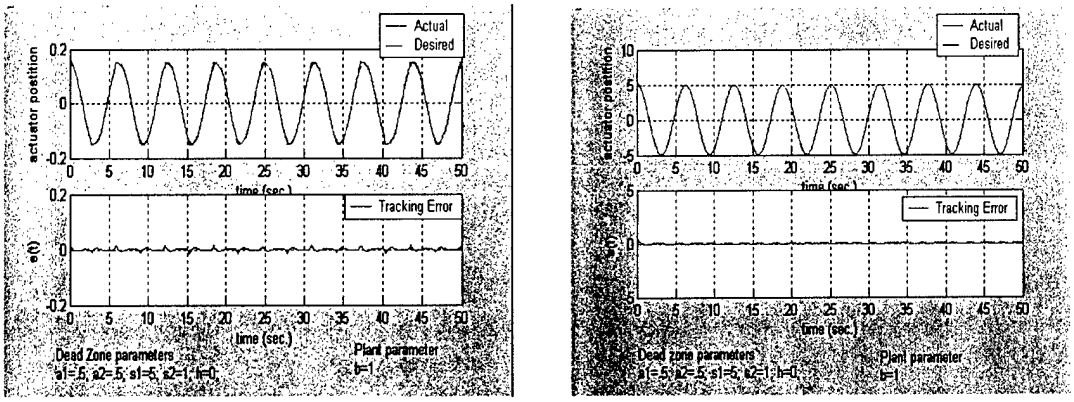


Figure 5-5. Plots Showing Tracking for Small and Moderate Sized Input

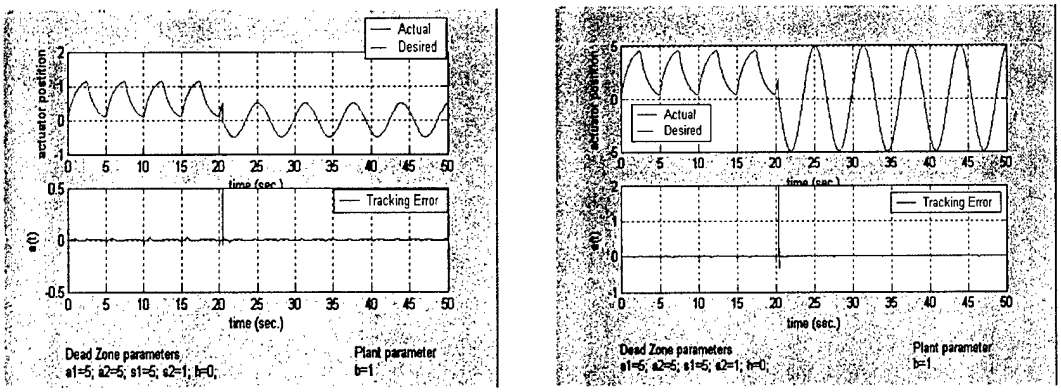


Figure 5-6. Plots Showing Tracking for Small and Moderate Sized Inputs Whose Characteristic Shape is Time Varying

The internal-model controller formulation is now applied to the Matlab MMT Model shown in Figure 5-7. The highlighted block is the internal model, and comparison is done for cases where the commanded signal is fed directly into the actuator and where the commanded signal is fed into the internal and the output of the internal model drives the actuator.

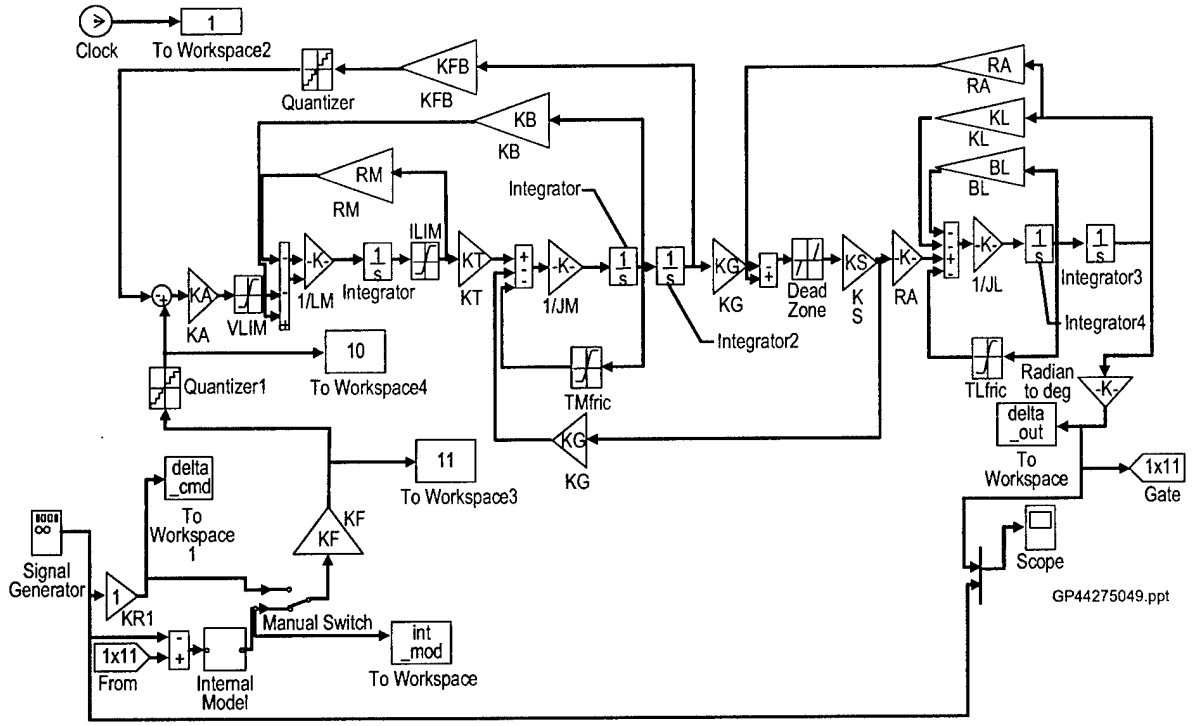


Figure 5-7. Matlab MMT Model with Internal Model-Based Controller

The nominal system (no internal model) suffers from limited response for 'small' inputs due to the piecewise linear dynamics in the actuator. It will be shown that the internal model greatly improves the response in these cases as well as maintains satisfactory response for larger inputs. The problem is approached in the same way that the previous problem was formulated and solved. The form of the IMPNOR scheme is the same with different regulating gains. However, in this case, no stabilizing controller was necessary due to the strong stability properties of the actuator dynamics.

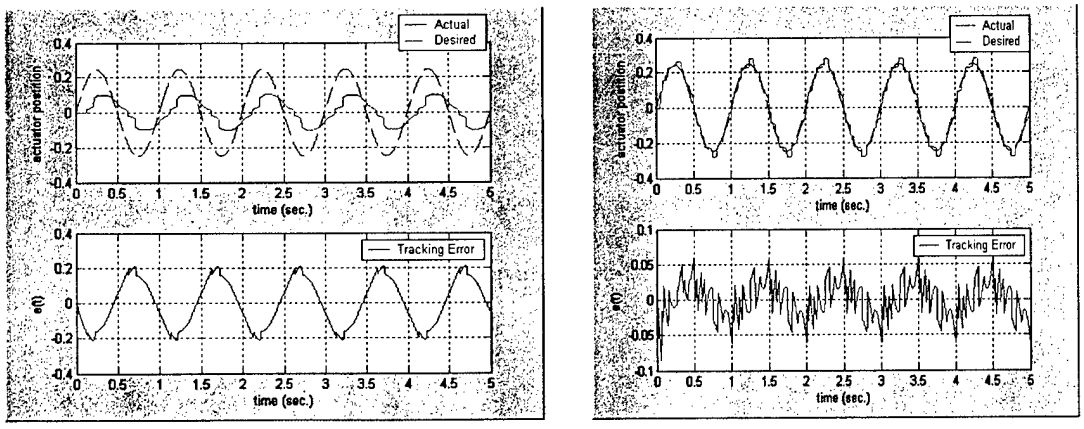


Figure 5-8. Plots Showing Tracking for Small Reference Input without IMPNOR Scheme (left) and with IMPNOR Scheme (right)

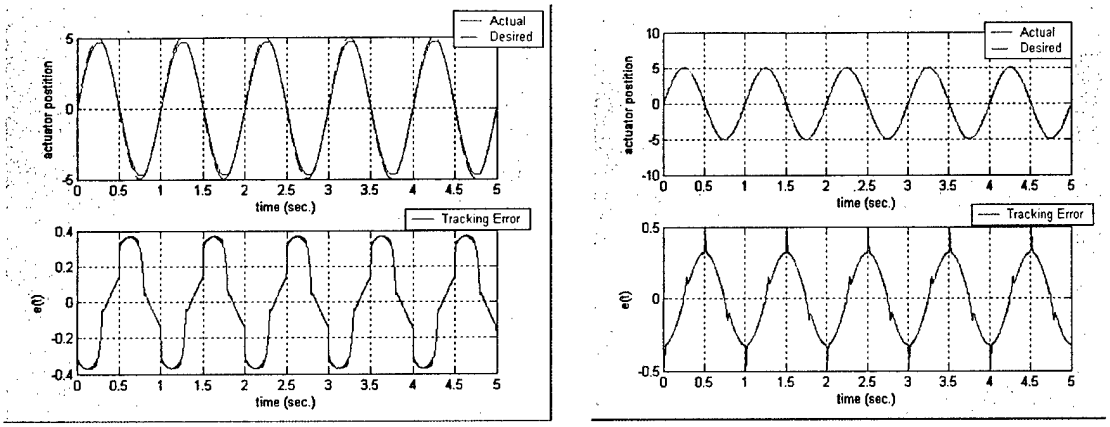


Figure 5-9. Plots Showing Tracking for Larger Reference Input without IMPNOR Scheme (left) and with IMPNOR Scheme (right)

The spike-like behavior observed for the IMPNOR scheme can be removed by filtering the feedback signal x(t) without adversely affecting the overall performance. The most important feature is the improvement gained for 'small' inputs. In fact, if the amplitude of the reference is reduced to 0.15 units, then the system without the IMPNOR scheme produces no output due to the dead zone. On the other hand, when the IMPNOR scheme is in the loop, some tracking is recovered. See Figure 5-10.

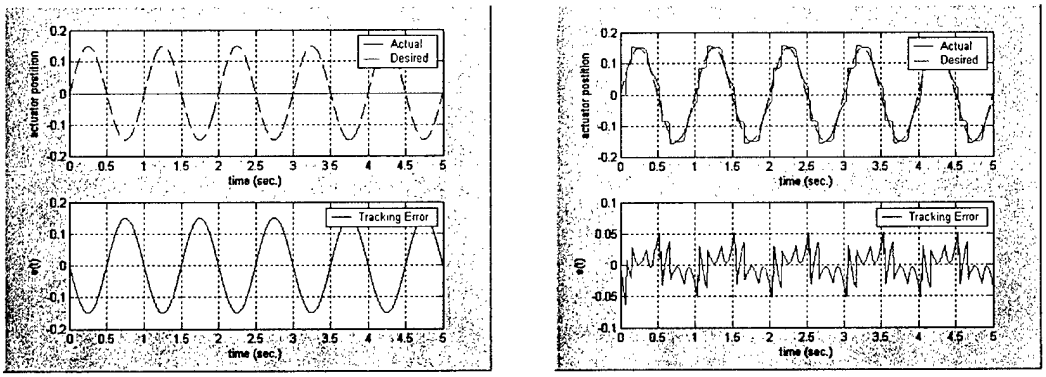


Figure 5-10. Plots Showing No Tracking for Very Small Input without IMPNOR Scheme (left) and Approx. Tracking with IMPNOR Scheme (right)

5.4 Summary

The IMPNOR method is able to maintain tracking in the presence of dead zone and Coulomb friction. The IMPNOR scheme realizes a controller that generates a copy of the required inverse dead zone dynamics needed to cause the tracking error to asymptotically approach zero. The main benefit lies in the ability to track small amplitude signals in spite of the dead zone. Therefore, the overall performance of the baseline system is improved noticeably when the internal model is added.

6.0 DEPARTURE ANALYSIS TOOL FOR UCAVS

Extensive investigations of robustness analysis for linear systems were done during the 1980's, which brought together many useful results with the goal of developing a practical robustness analysis tool. Two early significant developments, upon which many robustness analysis techniques rely, may be the structured singular value analysis by Doyle et. al. [29], and the stability margin analysis with real parameter variations by DeGaston and Safonov [30]. The structured singular value analysis provides a way to predict the exact stability margin if the system has dimension less than four, and the lower bound of the stability margin if the system has dimension greater than or equal to four in the presence of uncertain complex parameters. The method developed by DeGaston and Safonov provides a way to predict the exact stability margin in the presence of uncertain real parameters. At the industry level, Wise implemented the latter method successfully into a numerical tool called ROBUSTR [31], and performed robustness analysis for various missiles and aircraft applications.

Analysis of nonlinear systems, on the other hand, has stayed mainly at the theoretical level. A large body of work has been devoted to extend the linear analysis techniques to nonlinear systems. However, computational methods have not been developed to a level comparable to their linear counterparts. In predicting aircraft departure, much computation is required to perform nonlinear robustness analysis when linear analysis can not be used to approximate the highly nonlinear unsteady aerodynamic characteristics.

Murray et. al. presented a numerical method for robustness analysis for nonlinear systems [10]. Murray's method is briefly summarized below in order to establish a basis of our approach toward the development of a numerically efficient tool for nonlinear robustness analysis.

6.1 Problem Formulation

Consider the nonlinear system given by

$$\begin{aligned} \dot{x} &= F(x,u,v,\delta,U,t) \\ x(t_0) &= x_0 \end{aligned} \tag{6.1}$$

where F represents a smooth nonlinear function of a state x, a set of signals u corresponding to external disturbances, a set of signals v corresponding to the effect of components which are not independent of the other signals, but exact dependence is not known, a set of parameters δ, and a set of nominal commands U responsible for steering the nominal system along the nominal trajectory. The trajectory is described by a set of coordinates Y, given by a smooth function G of the same variables

$$Y = G(x,u,v,\delta,U,t) \tag{6.2}$$

The nominal trajectory Y_n is denoted by

$$Y_n = G(x,0,0,\delta_n,U,t) \tag{6.3}$$

where δ_n are the nominal values for the parameters.

Given an uncertain nonlinear system given by the equation (6.1) and the nominal command signal U, with initial conditions satisfying

$$c_i \le x_i(t_0) \le C_i \qquad i = 1,2,...,n,$$

the problem is to determine the maximum value of the norm of the error signal

$$\|y\| = \|G(x,u,v,\delta,U,t) - Y_n\| \tag{6.4}$$

subject to the constraints

$$\|u_i\|_2 = N_i \qquad i = 1,2,...,m$$

$$\|v_i\|_2 = \|z_i\|_2 \qquad i = 1,2,...,p$$

$$d_i \le \delta_i \le D_i \qquad i = 1,2,\ldots,r$$

where z is a variable representing uncertainty output given by a smooth function H

$$z = H(x,u,v,\delta,U,t).$$

6.2 Necessary Conditions

The necessary condition for a local maximum of the problem stated in the previous section is given by the following theorem:

Theorem: For a dynamical system described by the equations:

$$\dot{x} = f(x,u,t)$$

$$x(0) \text{ given}$$

$$t_0 \le t \le t_f$$

given a performance index of the form

$$J = \int_{t_0}^{t_f} L(x,u,t)dt$$

and a restriction on the final state

$$G(x(t_f)) = c,$$

if the signal u_0 achieves an extremum of J, then there exists a vector of constants ζ and a solution to the two-point boundary problem

$$\dot{x} = f(x,u,t)$$

$$\dot{\lambda} = -\left(\frac{\partial f}{\partial x}\right)^t \lambda - \left(\frac{\partial L}{\partial x}\right)^t$$

$$0 = \left(\frac{\partial L}{\partial u}\right) + \left(\frac{\partial f}{\partial u}\right)^t \lambda$$

with boundary conditions:

$$x(0) \text{ given}$$

$$\lambda(t_f) = \left(\frac{\partial G}{\partial x(t_f)} \right)' \varsigma$$

Furthermore, if these conditions are met we will have

$$\lambda(t_i) = \frac{\partial J}{\partial x(0)} \quad .$$

6.3 Implementation Issue

Although the technique discussed in [10] is intended to demonstrate a numerically efficient method for robustness analysis for nonlinear system, the technique requires to solve the two-point boundary problem as discussed above, which is normally time consuming. Boeing visited Dr. Murray at California Institute of Technology (Caltech) to find out whether or not further research that advances the technique has been done. After the discussion, we learned Dr. Murray and his research group at Caltech continue to investigate stability and robustness analysis method, however, the research direction has been changed, and any significant advancement of the technique of interest has not been made yet. Therefore, we decided not to investigate the technique any further toward the implementation of the departure analysis tool at this time.

subject to a system

$$\begin{aligned} \dot{X} &= f(X,u,t), \quad t_0 \le t \le t_f \\ X(t_0) &= X_0 \end{aligned} \tag{7.2}$$

where h is at least a C^1 function of X. Then by solving the Euler-Lagrange equation, we have

$$\begin{aligned} \dot{\lambda} &= -\left(\frac{\partial f}{\partial X}\right)^t \lambda - \left(\frac{\partial L}{\partial X}\right)^t \\ 0 &= \left(\frac{\partial L}{\partial u}\right) + \left(\frac{\partial f}{\partial u}\right)^t \lambda \end{aligned} \tag{7.3}$$

with boundary conditions:

$$\begin{aligned} X(t_0) &= X_0 \\ \lambda(t_f) &= \left(\frac{\partial h}{\partial X(t_f)}\right)^t \end{aligned} \tag{7.4}$$

Now, to solve our problem of interest, we set

$$L(X,u,t) = \frac{1}{2}u^2 \quad \text{and} \quad h(X(t_f)) = \frac{1}{2||X(t_f)||^2} \tag{7.5}$$

where $t_f \in [0, T]$ denotes the time instance at which the minimum separation occurs, i.e.,

$$||X(t_f)|| = \min_{0 \le t \le T} ||X(t)||, \tag{7.6}$$

and where $u(t) \in R$ denotes a roll rate command, and $X(t) \in R^3$ represents a relative aircraft position. Then Eq. (7.3) yields

$$\dot{\lambda} = -\left(\frac{\partial f}{\partial X}\right)^t \lambda$$

with the terminal condition

$$\lambda(t_f) = -\frac{X(t_f)}{||X(t_f)||^4}, \tag{7.8}$$

and

$$u(t) = -\left(\frac{\partial f}{\partial u}\right)^t \lambda(t) \tag{7.9}$$

Since

$$\lambda(t) = -e^{-\int_t^{t_f} (\frac{\partial f}{\partial X})^t \, dt} \frac{x(t_f)}{||x(t_f)||^4} \tag{7.10}$$

the optimal control u^o

$$u^o(t) = \left(\frac{\partial f}{\partial u}\right)^t \left[e^{-\int_t^{t_f} (\frac{\partial f}{\partial X})^t \, dt} \frac{x(t_f)}{||x(t_f)||^4}\right] \tag{7.11}$$

By integrating $u^o(t)$ from t = t_0 to $t = t_f$, the optimal roll angle ϕ° is determined as

$$\begin{aligned} \phi^o &= \int_{t_0}^{t_f} u^o(t)\, dt \\ &= \left[\int_{t_0}^{t_f} \left(\frac{\partial f}{\partial u}\right)^t dt\right]\left[e^{-\int_t^{t_f} (\frac{\partial f}{\partial X})^t \, dt} \frac{x(t_f)}{||x(t_f)||^4}\right] \end{aligned} \tag{7.12}$$

This roll angle can be shared between two aircraft such that

$$\int_{t_0}^{t_f} u_1(t) + \int_{t_0}^{t_f} u_2(t) = \phi^o(t) \tag{7.13}$$

where $u_i(t)$ i = 1, 2 is the roll rate command for aircraft 1 and 2, respectively.

To calculate the optimal roll angle ϕ°, the knowledge about t_f is required a priori, which is not known. Therefore, we need to find t_f as the fixed point solution of the following set of equations:

$$\begin{aligned} &\dot{X} = f(X, u^o, t), \quad X(t_0) = X_0 \\ &u^o(t) = \left(\frac{\partial f}{\partial u}\right)^t \left[e^{-\int_t^{t_f} (\frac{\partial f}{\partial X})^t \, dt} \frac{x(t_f)}{||x(t_f)||^4}\right] \\ &||X(t_f)|| = \min_{0 \le t \le T} ||X(t)|| \end{aligned} \tag{7.14}$$

To specify the smooth function $f(X, u, t)$, we need an aircraft model.

7.2 Simple Generic Aircraft Model

In the earth-fixed coordinate system, a generic aircraft dynamics can be derived as:

$$\begin{aligned} \dot{V}_t &= \frac{1}{m}(T_h \cos\alpha - D) - g\sin\gamma \\ \dot{\psi} &= \frac{N_z g \sin\varphi}{V_t \cos\gamma} \\ \dot{\gamma} &= \frac{g}{V_t}(N_z \cos\varphi - \cos\gamma) \\ \dot{\varphi} &= p + \dot{\psi}\sin(\gamma + \alpha) \end{aligned} \tag{7.15}$$

Here, V_t, ψ, γ, φ, α, T_h, D, p, N_z, g and m denote the airspeed, heading angle, flight path angle, bank angle, angle of attack, thrust, drag, roll rate, normal acceleration, gravitational acceleration, and aircraft mass, respectively. The normal acceleration N_z and the incremental roll angle $\Delta\zeta$ are commanded to generate the escape flight path. In addition, the responses of the roll rate p and normal acceleration N_z to their command inputs are approximated by the low order

equivalent systems, which are described by first and second order responses, respectively. That is:

$$\dot{p} = -\frac{1}{\tau}(p - p_{cmd}) \tag{7.16}$$

and

$$\ddot{N}_z + 2\zeta\omega\dot{N}_z + \omega^2(N_z - N_{z,cmd}) = 0 \tag{7.17}$$

where τ, ζ and ω respectively represent a roll mode time constant, damping ratio, and short period frequency. $N_{z,cmd}$ and p_{cmd} are maximum allowable normal acceleration command and roll rate command, respectively. The maximum allowable roll rate command is integrated over a certain period of time to meet the commanded roll angle. In the projected escape flight path calculation, the throttle setting is assumed to be constant. In other words, the thrust T_h is assumed to be constant throughout the projected escape flight path. In addition, the angle of attack α is calculated by solving

$$N_z = \frac{L + T_h \sin\alpha}{mg} \tag{7.18}$$

where L denotes the lift. The lift L and drag D are calculated by looking up a table for each flight condition using a generic aircraft model.

Then the velocity component of the aircraft in the earth-fixed coordinated system can be calculated by

$$\begin{aligned} \dot{x} &= V_t \cos\gamma \cos\psi \\ \dot{y} &= V_t \cos\gamma \sin\psi \\ \dot{z} &= V_t \sin\gamma \end{aligned} \tag{7.19}$$

By integrating equation (7.19) with respect to time, the projected aircraft position ($x(t)$, $y(t)$, $z(t)$) at time t can be calculated.

7.3 Summary

An analytic solution to the Automatic Air Collision Avoidance problem was discussed. The current method requires a priori knowledge of the time at minimum separation. This necessitates the fix-point iteration before advancing the time. Therefore it is not viable for the real-time operation. However, the objective of the current research is not to develop a real-time method (for now at least) for the air collision avoidance problem, but rather to search for the analytic optimal solution. This would provide an insight into what needs to be improved for some real-time methods based on numerical approach, particularly when the optimality of the solution is questionable. Of particular interest is to evaluate how the optimal escape maneuver calculated mathematically compares with the escape maneuver a pilot considers the best, and to analyze what needs to be done in order to fill the gap between them.

8.0 REFERENCES

[1] A. Isidori, "Nonlinear Control Systems," Springer-Verlag, New York, NY, 3rd edition, 1995.

[2] A. Isidori and C. I. Byrnes, "Output Regulation of Nonlinear Systems," IEEE Transaction on Automatic Control, AC-25, pp131-140,1990.

[3] A. Serrani and A. Isidori, "Global Output Regulation for a Class of Nonlinear Systems," 38th IEEE CDC Conference, 1999.

[4] A. Serrani, "Robust and Adaptive Output Regulation of Nonlinear Systems," Technical Report, Dept. of Systems Science and Mathematics, Washington University.

[5] A. Serrani and A. Isidori, "Robust Output Regulation for a Class of Non-minimum Phase Systems," 37th IEEE CDC Conference, 1998.

[6] S. Bittanti, F. Lorito, S. Strada, "An LQ approach to active control of vibrations in helicopters," *Trans. ASME, J. Dynamical Systems, Measurement and Control,* vol. 118, pp. 482-488,1996.

[7] C.I. Byrnes, A. Isidori, "Bifurcation analysis of the zero dynamics and the practical stabilization of nonlinear minimum-phase systems," *Asian J. of Control,* to appear (2002).

[8] B.A. Francis, W.M. Wonham, "The internal model principle of control theory," *Automatica*, vol. 12, pp. 457-465, 1977.

[9] R. Shoureshi, L. Brackney, N. Kubota, G. Batta, "A modern control approach to active noise control," *Trans. ASME J. Dynamical Systems, Measurement and Control,* vol. 115, pp. 673-678, 1993.

[10] J. E. Tierno, R. M. Murray, and J. C. Doyle, "Numerically Efficient Robustness Analysis of Trajectory Tracking for Nonlinear Systems," AIAA Journal of Guidance, Control, and Dynamics, Nov., 1995.

[11] C. I. Byrnes, D. S. Gilliam, A. Isidori, Y. Ikeda, L Marconi, "Internal Model Based Design for the Suppression of Harmonic Disturbances," Directions in Mathematical Systems Theory and Optimization, Lecture Notes in Control and Information Sciences 286, Springer-Verlag, Berlin Hedelberg, 2003.

[12] A. Melikyan, N. Hovakimyan, and Y. Ikeda, "Dynamic Programming Approach to a Minimum Distance Optimal Control Problem," 42nd IEEE Conference on Decision and Control, December, 9-12, 2003.

[13] K. S. Narendra, A. M. Annaswamy, Stable Adaptive Control, Prentice Hall, 1989.

[14] S. S. Sastry, M. Bodson, Adaptive Control: Stability, Convergence and Robustness, Prentice Hall, 1989.

[15] J. J. Slotine, W. Li, Applied Nonlinear Control, Prentice Hall, New Jersey, 1991.

[16] K. A. Wise, J. L. Sedwick, "Successive Approximation Solution to the Hamilton-Jacobi-Isaacs Equation, Proc of the 33rd IEEE CDC, Orlando FL, Dec. 1994, pp. 1387-1391.

[17] K. A. Wise, D. Broy, "Agile Missile Dynamics and Control," *Journal of Guidance, Control, and Dynamics*, Vol. 21, No. 3, 1998, pp. 441-449.

[18] K. A. Wise, "Fighter Aircraft Control Challenges and Technology Transition," *Systems and Control in the Twenty-First Century*, Birkhauser, 1996.

[19] K. A. Wise, J. L. Sedwick, "Nonlinear Control Of Agile Missiles Using State Dependent Riccati Equations, Proc. Of the 1997 ACC, Albuquerque NM, June 1997, pp. 379-380.

[20] K. A. Wise, D. Broy, "Agile Missile Dynamics and Control," AIAA paper No. 96-3912, presented at the AIAA GNC Conference, San Diego CA, August 1996.

[21] F. Scarselli, A. C. Tsoi, "Universal Approximation using Feedforward Neural Networks: A Survey of some existing methods and Some New Results," Neural Networks, vol. 11, No. 1, pp. 15-37.

[22] S. Haykin, "Neural Networks: A Comprehensive Foundation", - 2nd edition, Prentice Hall Inc., 1999.

[23] J. Hertz, A. Krogh, R. G. Palmer, "Introduction to the Theory of Neural Computation", Addison-Wesley Publishing Company, 1991.

[24] E. Lavretsky, N. Hovakimyan, A. Calise, "Adaptive Extremum Seeking Control Design", American Control Conference, May 2003.

[25] E. Lavretsky, "F/A-18 Autonomous Formation Flight Control System Design", AIAA-2002-4757, AIAA Guidance, Navigation and Control Conference, August 2002, Monterey, CA.

[26] B. Nguyen, A. Barfield, Y. Ikeda, C. Carlsson, "Preliminary Simulation Predictions of Nuisance Criteria for an Automatic Air Collision Avoidance System," The 9th St. Petersburg International Conference on Integrated Navigation Systems, May, 2001.

[27] K. A. Wise, J. Brinker, "Miniaturized Munition Technology Actuator Analysis," McDonnell Douglas Aerospace Technical Report.

[28] J. E. Tao, P. Kokotovic, "Adaptive Control of Systems," John Wiley & Sons, Inc., 1996.

[29] J. C. Doyle, J. E. Wall, G. Stein, "Performance and Robustness Analysis for Structured Uncertainty," 21st IEEE CDC Conference, 1982.

[30] R. R. Degaston and Safonov, "Exact Calculation of the Multiloop Stability Margin," IEEE Transaction on Automatic Control, Vol. 33, No.2, pp 156 – 171, Feb, 1988.

[31] K.A. Wise, "Robustness Analysis Software for Real Parameter Variations: ROBUSTR," User Manual and Documentation & Software Listings, McDonnell Douglas Aerospace-East, April, 1993.

[32] Byrnes, C.I., Delli Priscoli, F., Isidori, A., "Output Regulation of Uncertain Nonlinear Systems", *Birkhauser*, 1997.

[33] Francis, B.A. Wonham, W.M., "The Internal Model Principle of Control Theory", *Automatica*, 12: 457-465, 1976.

APPENDIX A
PROJECTION OPERATOR

Definition A.1: Subset $\Pi \subset R^n$ is convex if

$$\left[\forall x, y \in \Pi \subset R^n\right] \Rightarrow \left[\lambda x + (1-\lambda) y = z \in \Pi\right], \ \forall 0 \le \lambda \le 1 \tag{A.1}$$

Relation (A.1) states that if two points belong to a convex subset Π then all the points on the connecting line must also belong to the same subset Π.

Definition A.2: Function $f: R^n \to R$ is convex if

$$f\left(\lambda x + (1-\lambda) y\right) \le \lambda f(x) + (1-\lambda) f(y), \ \forall 0 \le \lambda \le 1 \tag{A.2}$$

Relation A.2 is illustrated in Figure A-1, and states that graph of a convex function must be located below the straight line that connects any two corresponding function values.

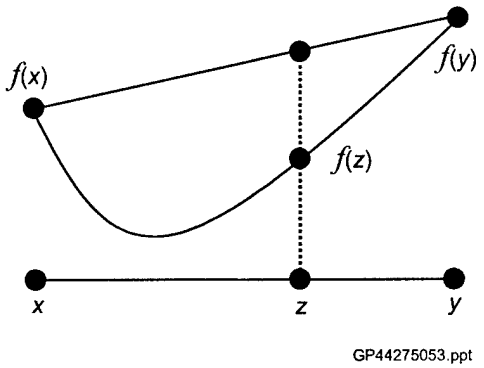


Figure A-1. Convex Function

Statement A.1: Let $f(x): R^n \to R$ be convex function. Then for any constant $\delta > 0$ the subset $\Pi_\delta = \left\{\theta \in R^n \,\middle|\, f(\theta) \le \delta\right\}$ is convex.

Proof: Let $\theta_1, \theta_2 \in \Pi_\delta$. Then $f(\theta_1) \le \delta$ and $f(\theta_2) \le \delta$. Since $f(x)$ is convex then for any $0 \le \lambda \le 1$

$$f\left(\underbrace{\lambda\theta_1 + (1-\lambda)\theta_2}_{\theta}\right) \le \lambda \underbrace{f(\theta_1)}_{\le\delta} + (1-\lambda)\underbrace{f(\theta_2)}_{\le\delta} \le \lambda\delta + (1-\lambda)\delta = \delta$$

Therefore $f(\theta) \le \delta$ and, consequently, $\theta \in \Pi_\delta$ which completes the proof.

Statement A.2: Let $f(x): R^n \to R$ be a continuously differentiable convex function. Choose a constant $\delta > 0$ and consider the subset $\Pi_\delta = \left\{\theta \in R^n \,\middle|\, f(\theta) \le \delta\right\} \subset R^n$. Let $\theta^* \in \Pi_\delta$ and assume

that $f(\theta^*) < \delta$, (i.e., θ^* is not on the boundary of Π_δ). Also, let $\theta \in \Pi_\delta$ and assume that $f(\theta) = \delta$, (i.e., (θ is on the boundary of Π_δ). Then the following inequality takes place:

$$(\theta^* - \theta)^T \nabla f(\theta) \le 0 \tag{A.3}$$

where $\nabla f(\theta) = \left(\frac{\partial f(\theta)}{\partial \theta_1} \quad \dots \quad \frac{\partial f(\theta)}{\partial \theta_n} \right)^T \in R^n$ is the gradient vector of f evaluated at θ.

Relation (A.3) is illustrated on Figure A-2. It shows that the gradient vector evaluated at any boundary point of a convex set always points away from the set.

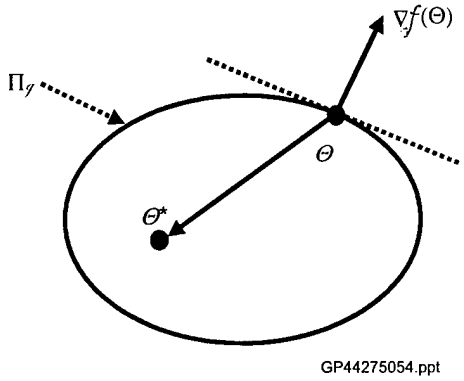


Figure A-2. Gradient and Convex Set

Proof: Since $f(x)$ is convex then

$$f(\lambda\theta^* + (1-\lambda)\theta) \le \lambda f(\theta^*) + (1-\lambda) f(\theta)$$

or equivalently:

$$f(\theta + \lambda(\theta^* - \theta)) \le f(\theta) + \lambda(f(\theta^*) - f(\theta))$$

Then for any nonzero $0 < \lambda \le 1$:

$$\frac{f(\theta + \lambda(\theta^* - \theta)) - f(\theta)}{\lambda} \le f(\theta^*) - f(\theta) < \delta - \delta = 0$$

Taking the limit as $\lambda \to 0$ yields relation (A.3) and completes the proof.

Suppose that θ, the "true" parameter vector, belongs to a *convex* set Π_0

$$\Pi_0 = \{\theta \in R^n \mid f(\theta) \le 0\} \tag{A.4}$$

Introduce another convex set:

$$\Pi_1 = \{\theta \in R^n \mid f(\theta) \le 1\} \tag{A.5}$$

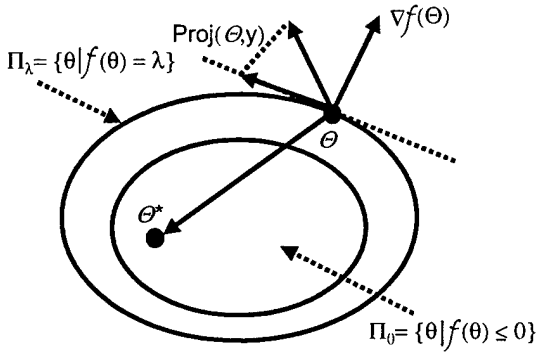
It is obvious that $\Pi_0 \in \Pi_1$. We may now define the *projection operator*:

$$\mathrm{Proj}(\theta, y) = \begin{cases} y & \text{, if } f(\theta) \le 0 \\ y & \text{, if } \left[f(\theta) \ge 0 \wedge y^T \nabla f(\theta) \le 0 \right] \\ y - \dfrac{\nabla f(\theta)\left(\nabla f(\theta)\right)^T}{\left\| \nabla f(\theta) \right\|^2} y f(\theta) & \text{, if not.} \end{cases} \tag{A.6}$$

or, equivalently:

$$\mathrm{Proj}(\theta, y) = \begin{cases} y - \dfrac{\nabla f(\theta)\left(\nabla f(\theta)\right)^T}{\left\| \nabla f(\theta) \right\|^2} y f(\theta) & \text{, if } \left[f(\theta) > 0 \wedge y^T \nabla f(\theta) \right] > 0 \\ y & \text{, if not} \end{cases} \tag{A.7}$$

Namely, $\mathrm{Proj}(\theta, y)$ does not alter the vector y if θ belongs to the convex set Ω_0 defined in (A.4). In the set $\{0 \le f(\theta) \le 1\}$, the projection operator subtracts a vector normal to the boundary $\{f(\theta) = \lambda\}$ from y so that we get a smooth transition from the original vector field y for $\lambda = 0$ to a tangent to the boundary vector field for $\lambda = 1$. The projection operator concept is illustrated on Figure A-3.



GP44275055.ppt

Figure A-3. Projection Operator

Using Statement (A.2) and inequality (A.3), we get the following important property of the projection operator:

$$\left(\theta^* - \theta\right)^T \left(y - \mathrm{Proj}(\theta, y)\right) = \left\{ \begin{array}{l} 0, \text{ if } f(\theta) \le 0 \\ 0, \text{ if } f(\theta) \ge 0 \text{ and } y^T \nabla f(\theta) \le 0 \\ \dfrac{\underbrace{\left(\theta^* - \theta\right)^T \nabla f(\theta)}_{\le 0} \underbrace{\left(\nabla f(\theta)\right)^T y}_{\ge 0}}{\left\|\nabla f(\theta)\right\|^2} \underbrace{f(\theta)}_{=\lambda \ge 0}, \text{ if not} \end{array} \right\} \le 0 \qquad \text{(A.8)}$$

or, equivalently

$$\left(\theta - \theta^*\right)^T \left(\mathrm{Proj}(\theta, y) - y\right) \le 0 \qquad \text{(A.9)}$$

Based on (A.6), we can now define the projection operator when both Y and Θ are matrices of the same dimensions, that is: $Y = \begin{pmatrix} y_1 & \dots & y_N \end{pmatrix} \in R^{n \times N}$ and $\Theta = \begin{pmatrix} \theta_1 & \dots & \theta_N \end{pmatrix} \in R^{n \times N}$

$$\mathrm{Proj}(\Theta, Y) = \begin{pmatrix} \mathrm{Proj}(\theta_1, y_1) & \dots & \mathrm{Proj}(\theta_N, y_N) \end{pmatrix} \qquad \text{(A.10)}$$

Relation (A.10) states that for matrices the projection operator is defined column-wise.

Finally, we show that the use of the projection operator (4.29) makes the trace term in the Lyapunov function derivative (4.28) semi-negative.

$$trace\left(\Delta\Omega^T \left(\underbrace{\Psi(x,u) e^T P B_1}_{-Y} + \underbrace{\Gamma^{-1} \dot{\Omega}}_{\mathrm{Proj}(\Omega, Y)} \right) \Lambda \right) = \sum_{j=1}^{m} \underbrace{\Delta\Omega_j^T \left(\mathrm{Proj}\left(\Omega_j, Y_j\right) - Y_j \right)}_{\le 0} \underbrace{\lambda_j}_{\ge 0} \le 0 \qquad \text{(A.11)}$$

In (A.11), Ω_j and Y_j denote j-th columns of Ω and Y, correspondingly, while λ_j represents the j-th non-negative element on the diagonal of Λ. From (A.11), we get the same adaptation law as specified in (4.29).

Next, we show how to define the convex function $f(\cdot)$ that in turn defines the convex set Π_δ. Both definitions are based on the known upper bound $\left\|\Omega_j\right\| \le \Omega_j^{\max}$ that is imposed column-wise. For each column of Ω, choose a positive constant ε_j that represents a desired projection tolerance, and define convex function:

$$f_j = f\left(\Omega_j\right) = \frac{\left\|\Omega_j\right\|^2 - \left(\Omega_j^{\max}\right)^2}{\varepsilon_j} = \frac{\left(\Omega_j^T \Omega_j\right) - \left(\Omega_j^{\max}\right)^2}{\varepsilon_j} \qquad \text{(A.12)}$$

and convex set

$$\Pi_\delta^j = \left\{ \Omega_j \,\middle|\, f_j \le \delta_j \right\} \qquad \text{(A.13)}$$

Then for each $j = 1, \ldots, m$:

$$\begin{cases} \Pi_0^j = \left\{ \Omega_j \,\middle|\, f_j \le 0 \right\} = \left\{ \Omega_j \,\middle|\, \left\| \Omega_j \right\| \le \Omega_j^{\max} \right\} \\ \Pi_1^j = \left\{ \Omega_j \,\middle|\, f_j \le 1 \right\} = \left\{ \Omega_j \,\middle|\, \left\| \Omega_j \right\|^2 \le \left(\Omega_j^{\max} \right)^2 + \varepsilon_j \right\} \end{cases} \tag{A.14}$$

Adaptive laws (4.29) require computation of the gradient of the convex function (A.12).

$$\nabla f\left(\Omega_j \right) = \frac{2}{\varepsilon_j} \Omega_j \tag{A.15}$$

Finally, it should be noted that the adaptation with projection ensures the uniform boundedness of each column of the adaptive matrix Ω forward in time:

$$\left\| \Omega_j (t) \right\|^2 \le \left(\Omega_j^{\max} \right)^2 + \varepsilon_j, \forall 1 \le j \le m, \forall t \ge 0 \tag{A.16}$$

APPENDIX B
ADAPTATION USING SIGMOIDAL NEURAL NETWORKS

The universal approximation property of single-hidden-layer (SHL) neural network (NN) architectures states that a continuous scalar function $f(x): R^n \to R$ can be approximated by a SHL NN with sufficiently many neurons on a compact (closed and bounded) domain $D \subset R^n$ to within any given tolerance, [46]. SHL NN estimate of a scalar function f has the following form:

$$\hat{f}(x) = \sum_{i=1}^{N} w_i \, \sigma\left(v_i^T x + b_i\right) + c_0 \tag{B.1}$$

where N is the number of neurons in the hidden layer, $w \in R^N$ is the vector of output weights, $v_i \in R^n$ is the vector of inner-layer weights, and n is the dimension of $x \in R^n$. Moreover, the vector of thresholds is denoted by $b \in R^N$, c_0 represents the NN bias term. In (B.1), the scalar function $\sigma(z)$, called the sigmoid, models the output activity of each neuron. A commonly used sigmoid function is $\sigma(z) = \left(1 + e^{-z}\right)^{-1}$. As seen from (B.1), the SHL NN approximation is linear in the output weights w_i and the bias term c_0, but it is nonlinear in the inner-layer weights v_i and the thresholds b_i. In formal terms, the SHL NN universal approximation property can be stated as follows:

$$\forall \varepsilon > 0 \; \exists N, c_0 \in R, \exists w, b \in R^N, \exists v \in R^n : \left| f(x) - \hat{f}(x) \right| \le \varepsilon, \; \forall x \in D \tag{B.2}$$

Relation (B.2) can also be written in a matrix form:

$$\hat{f}(x) = w^T \vec{\sigma}\left(V^T x + b\right) + c_0 \tag{B.3}$$

where the vector of sigmoids $\vec{\sigma}$ and the n-by-N matrix V are defined below:

$$\vec{\sigma}\left(V^T x + b\right) = \left[\sigma\left(v_1^T x + b_1\right) \quad \dots \quad \sigma\left(v_N^T x + b_N\right)\right]^T \in R^N \tag{B.4}$$

$$V = \left[v_1 \quad \dots \quad v_N\right] \in R^{n \times N} \tag{B.5}$$

In order to not discriminate between the output weights w and the bias term c_0, the inner-layer weights v_i and the thresholds b, both the n-dimensional independent variable x and the N-dimensional vector of sigmoids σ are augmented with 1-s. Then the SHL NN approximation (B.3) can be written as:

$$\hat{f}(x) = \overline{w}^T \vec{\sigma}\left(\overline{V}^T \quad \overline{x}\right) \tag{B.6}$$

where

$$\overline{x} = \left[x^T \quad 1\right]^T \in R^{n+1} \tag{B.7}$$

$$\bar{V} = \begin{bmatrix} V^T & b \end{bmatrix}^T \in R^{(n+1)\times N} \tag{B.8}$$

$$\bar{w} = \begin{bmatrix} w^T & c_0 \end{bmatrix}^T \in R^{N+1} \tag{B.9}$$

$$\bar{\sigma}\left(\bar{V}^T \bar{x}\right) = \begin{bmatrix} \sigma\left(\bar{v}_1^T \bar{x}\right) & \dots & \sigma\left(\bar{v}_N^T \bar{x}\right) & 1 \end{bmatrix}^T \in R^{N+1} \tag{B.10}$$

Finally, generalizing the single-output SHL NN (B.6) to handle multi-output function approximations yields:

$$\hat{F}(x) = \bar{W}^T \bar{\sigma}\left(\bar{V}^T \bar{x}\right) \tag{B.11}$$

where the output weights matrix $\bar{W}$ has the number of rows equal to the dimension of the vector-function $F(x)$ that is being approximated on the compact domain $D \subset R^n$. Note that while x in the left hand side of (B.11) represents the original independent variable, the right hand side of (B.11) depends on the augmented variable $\bar{x}$ defined by (B.7). In what follows and without a loss of generality, the SHL NN approximation (B.11) will be written in the form:

$$\hat{F}(x) = W^T \bar{\sigma}\left(V^T \mu\right) \tag{B.12}$$

In (B.12), $\mu = \begin{bmatrix} x^T & 1 \end{bmatrix}^T \in R^{n+1}$ and $\bar{\sigma}$ represents the $(N+1)$ – dimensional vector of sigmoids augmented with a unity.

$$\bar{\sigma}\left(V^T \mu\right) = \left(\sigma\left(v_1^T x + b_1\right) \quad \dots \quad \sigma\left(v_N^T x + b_N\right) \quad 1\right)^T \tag{B.13}$$

To this end, consider the system error dynamics and approximate the matched uncertainty $K_0\left(x_p\right)$ using SHL NN with N_0 sigmoids in its inner layer. The SHL NN Universal Approximation Theorem [46] states that given an approximation tolerance $\varepsilon_0^* > 0$, and a compact set $X \subset R^n$, there must exist an integer N_0 and "true" constant weight matrices W and V such that for all $x_p \in X \subset R^n$:

$$K_0\left(x_p\right) = W^T \bar{\sigma}\left(V^T \underbrace{\begin{pmatrix} x_p \\ 1 \end{pmatrix}}_{\mu}\right) + \varepsilon_0\left(x_p\right) \tag{B.14}$$

and

$$\left\|\varepsilon_0\left(x_p\right)\right\| \le \varepsilon_0^* \tag{B.15}$$

Let

$$\hat{K}_0\left(x_p\right) = \hat{W}^T \bar{\sigma}\left(\hat{V}^T \mu\right) \tag{B.16}$$

represent the neural approximation of the uncertainty $K_0\left(x_p\right)$. Then the approximation error becomes:

$$\Delta K_0(x_p) = \hat{K}_0(x_p) - K_0(x_p) = \hat{W}^T \vec{\sigma}(\hat{V}^T \mu) - W^T \vec{\sigma}(V^T \mu) - \varepsilon_0(x_p) \tag{B.17}$$

The SHL NN weigh approximation errors are defined as:

$$\Delta W = \hat{W} - W, \quad \Delta V = \hat{V} - V \tag{B.18}$$

Using Taylor series expansion of the vector-function $\vec{\sigma}(V^T \mu)$ around the vector $\hat{V}^T \mu$, we get:

$$\vec{\sigma}(\hat{V}^T \mu) - \vec{\sigma}(V^T \mu) = \vec{\sigma}'(\hat{V}^T \mu) \Delta V^T \mu + O\left(\left\|\Delta V^T \mu\right\|^2\right) \tag{B.19}$$

where $\vec{\sigma}'(\hat{V}^T \mu) = \{d\vec{\sigma}(z)/dz\}_{z=\hat{V}^T \mu}$ and $O(z^2)$ denotes Taylor series expansion terms of order 2. In that case:

$$\begin{aligned}
&\hat{W}^T \vec{\sigma}(\hat{V}^T \mu) - W^T \vec{\sigma}(V^T \mu) = \hat{W}^T \vec{\sigma}(\hat{V}^T \mu) - W^T \vec{\sigma}(V^T \mu) \pm W^T \vec{\sigma}(\hat{V}^T \mu) \\
&= \Delta W^T \vec{\sigma}(\hat{V}^T \mu) + W^T \left(\vec{\sigma}(V^T \mu) - \vec{\sigma}(\hat{V}^T \mu)\right) \\
&= \Delta W^T \vec{\sigma}(\hat{V}^T \mu) + W^T \left(\vec{\sigma}'(\hat{V}^T \mu) \Delta V^T \mu + O\left(\left\|\Delta V^T \mu\right\|^2\right)\right) \\
&= \Delta W^T \vec{\sigma}(\hat{V}^T \mu) + W^T \vec{\sigma}'(\hat{V}^T \mu) \Delta V^T \mu + W^T O\left(\left\|\Delta V^T \mu\right\|^2\right)
\end{aligned} \tag{B.20}$$

Further manipulation of the terms in (B.20) leads to:

$$\begin{aligned}
&\hat{W}^T \vec{\sigma}(\hat{V}^T \mu) - W^T \vec{\sigma}(V^T \mu) = \\
&= \Delta W^T \vec{\sigma}(\hat{V}^T \mu) + (\hat{W} - \Delta W)^T \vec{\sigma}'(\hat{V}^T \mu) \Delta V^T \mu + W^T O\left(\left\|\Delta V^T \mu\right\|^2\right) \\
&= \Delta W^T \vec{\sigma}(\hat{V}^T \mu) + \hat{W}^T \vec{\sigma}'(\hat{V}^T \mu) \Delta V^T \mu - \Delta W^T \vec{\sigma}'(\hat{V}^T \mu) \Delta V^T \mu + W^T O\left(\left\|\Delta V^T \mu\right\|^2\right)
\end{aligned} \tag{B.21}$$

Substituting for $\Delta V = \hat{V} - V$ into the 3rd term in (B.21) yields:

$$\begin{aligned}
&\hat{W}^T \vec{\sigma}(\hat{V}^T \mu) - W^T \vec{\sigma}(V^T \mu) = \Delta W^T \vec{\sigma}(\hat{V}^T \mu) + \hat{W}^T \vec{\sigma}'(\hat{V}^T \mu) \Delta V^T \mu \\
&- \Delta W^T \vec{\sigma}'(\hat{V}^T \mu)(\hat{V} - V)^T \mu + W^T O\left(\left\|\Delta V^T \mu\right\|^2\right) \\
&= \Delta W^T \left(\vec{\sigma}(\hat{V}^T \mu) - \vec{\sigma}'(\hat{V}^T \mu) \hat{V}^T \mu\right) + \hat{W}^T \vec{\sigma}'(\hat{V}^T \mu) \Delta V^T \mu \\
&+ \underbrace{\Delta W^T \vec{\sigma}'(\hat{V}^T \mu) V^T \mu + W^T O\left(\left\|\Delta V^T \mu\right\|^2\right)}_{\Omega}
\end{aligned} \tag{B.22}$$

Using (B.17) the uncertainty approximation error $\Delta K_0(x_p)$ can be written as a function that linearly depends on the SHL NN weights estimation errors ΔW and ΔV:

$$\begin{aligned}\Delta K_0(x_p) &= \hat{K}_0(x_p) - K_0(x_p) = \hat{W}^T \bar{\sigma}(\hat{V}^T \mu) - W^T \bar{\sigma}(V^T \mu) - \varepsilon_0(x_p) \\ &= \Delta W^T \left(\bar{\sigma}(\hat{V}^T \mu) - \bar{\sigma}'(\hat{V}^T \mu)\hat{V}^T \mu\right) + \hat{W}^T \bar{\sigma}'(\hat{V}^T \mu)\Delta V^T \mu + \Omega - \varepsilon_0(x_p)\end{aligned} \tag{B.23}$$

Substituting (B.23) into the error dynamics equation

$$\begin{aligned}\dot{e} &= A_{ref}\, e + B_1 \Lambda \left(\underbrace{\left(K_x - K_x^*\right)^T}_{\Delta K_x} x + \underbrace{\left(K_u - K_u^*\right)^T}_{\Delta K_u} u - \Delta K_0(x_p) \right) \\ &= A_{ref}\, e + B_1 \Lambda \left(\Delta K_x^T x + \Delta K_u^T u - \Delta K_0(x_p) \right)\end{aligned}$$

yields:

$$\begin{aligned}\dot{e} &= A_{ref}\, e + B_1 \Lambda \left(\Delta K_x^T x + \Delta K_u^T u - \Delta K_0(x_p) \right) \\ &= A_{ref}\, e + B_1 \Lambda \left(\Delta K_x^T x + \Delta K_u^T u \right) - B_1 \Lambda \left(\Omega - \varepsilon_0(x_p) \right) \\ &- B_1 \Lambda \left(\Delta W^T \left(\bar{\sigma}(\hat{V}^T \mu) - \bar{\sigma}'(\hat{V}^T \mu)\hat{V}^T \mu\right) + \hat{W}^T \bar{\sigma}'(\hat{V}^T \mu)\Delta V^T \mu \right)\end{aligned} \tag{B.24}$$

To this end, consider the following Lyapunov function candidate:

$$\begin{aligned}V(e, \Delta K_x, \Delta K_r, \Delta \tilde{W}, \Delta V) &= e^T P e \\ &+ \text{trace}\left(\Delta K_x^T \Gamma_x^{-1} \Delta K_x \Lambda\right) + \text{trace}\left(\Delta K_u^T \Gamma_u^{-1} \Delta K_u \Lambda\right) \\ &+ \text{trace}\left(\Delta W^T \Gamma_W^{-1} \Delta W \Lambda\right) + \text{trace}\left(\Delta V^T \Gamma_V^{-1} \Delta V\right)\end{aligned} \tag{B.25}$$

In (B.25), P is the unique symmetric positive definite matrix solution of the algebraic Lyapunov equation:

$$A_{ref}^T P + P A_{ref} = -Q$$

In (B.25), $\Gamma_x, \Gamma_u, \Gamma_W, \Gamma_V$ are symmetric positive definite matrices of the corresponding dimensions. The time derivative of V along the trajectories of the error dynamics is:

$$\begin{aligned}\dot{V}(e, \Delta K_x, \Delta K_r, \Delta W, \Delta V) &= -e^T Q e \\ &+ 2 e^T P B_1 \Lambda \left(\Delta K_x^T x + \Delta K_u^T u \right) \\ &- 2 e^T P B_1 \Lambda \Delta W^T \left(\bar{\sigma}(\hat{V}^T \mu) - \bar{\sigma}'(\hat{V}^T \mu)\hat{V}^T \mu\right) \\ &- 2 e^T P B_1 \Lambda \hat{W}^T \bar{\sigma}'(\hat{V}^T \mu)\Delta V^T \mu \\ &+ 2\,\text{trace}\left(\Delta K_x^T \Gamma_x^{-1} \dot{\hat{K}}_x \Lambda\right) + 2\,\text{trace}\left(\Delta K_u^T \Gamma_u^{-1} \dot{\hat{K}}_u \Lambda\right) \\ &+ 2\,\text{trace}\left(\Delta W^T \Gamma_W^{-1} \dot{\hat{W}} \Lambda\right) + 2\,\text{trace}\left(\Delta V^T \Gamma_V^{-1} \dot{\hat{V}}\right) \\ &- 2 e^T P B_1 \Lambda \left(\Omega - \varepsilon_0(x_p) \right)\end{aligned} \tag{B.26}$$

Using the trace identity $x^T y = \text{trace}\left(y x^T\right)$ we further get:

$$\begin{aligned}
&\dot{V}\left(e, \Delta K_x, \Delta K_r, \Delta W, \Delta V\right) = -e^T Q e \\
&+2\,\text{trace}\left(\Delta K_x^T \left(\Gamma_x^{-1} \dot{\hat{K}}_x + x e^T P B_1\right)\Lambda\right) \\
&+2\,\text{trace}\left(\Delta K_u^T \left(\Gamma_u^{-1} \dot{\hat{K}}_u + u e^T P B_1\right)\Lambda\right) \\
&+2\,\text{trace}\left(\Delta W^T \left(\Gamma_W^{-1} \dot{\hat{W}} - \left(\vec{\sigma}\left(\hat{V}^T \mu\right) - \vec{\sigma}'\left(\hat{V}^T \mu\right)\hat{V}^T \mu\right) e^T P B_1\right)\Lambda\right) \\
&+2\,\text{trace}\left(\Delta V^T \left(\Gamma_V^{-1} \dot{\hat{V}} - \mu e^T P B_1 \Lambda \hat{W}^T \vec{\sigma}'\left(\hat{V}^T \mu\right)\right)\right) \\
&-2 e^T P B_1 \Lambda\left(\Omega - \varepsilon_0\left(x_p\right)\right)
\end{aligned} \tag{B.27}$$

The following adaptive laws are chosen:

$$\begin{cases}
\dot{\hat{K}}_x = \Gamma_x\,\text{Proj}\left(\hat{K}_x, -x e^T P B_1\right) \\
\dot{\hat{K}}_u = \Gamma_u\,\text{Proj}\left(\hat{K}_u, -u e^T P B_1\right) \\
\dot{\hat{W}} = \Gamma_W\,\text{Proj}\left(\left(\vec{\sigma}\left(\hat{V}^T \mu\right) - \vec{\sigma}'\left(\hat{V}^T \mu\right)\hat{V}^T \mu\right) e^T P B_1\right) \\
\dot{\hat{V}} = \Gamma_V\,\text{Proj}\left(\mu e^T P B_1 \hat{W}^T \vec{\sigma}'\left(\hat{V}^T \mu\right)\right)
\end{cases} \tag{B.28}$$

Substitution of (B.28) into (B.27) yields:

$$\begin{aligned}
&\dot{V}\left(e, \Delta K_x, \Delta K_r, \Delta W, \Delta V\right) \\
&= -e^T Q e + 2\,\text{trace}\left(\Delta V^T \mu e^T P B_1 \left(I_m - \Lambda\right)\hat{W}^T \vec{\sigma}'\left(\hat{V}^T \mu\right)\right) - 2 e^T P B_1 \Lambda\left(\Omega - \varepsilon_0\left(x_p\right)\right) \\
&= -e^T Q e + 2 e^T P B_1 \left(\left(I_m - \Lambda\right)\hat{W}^T \vec{\sigma}'\left(\hat{V}^T \mu\right)\Delta V^T \mu - \Lambda\left(\Omega - \varepsilon_0\left(x_p\right)\right)\right) \\
&\le -\lambda_{\min}\left(Q\right)\|e\|^2 + 2\|e\|\|P B_1\|\left(\left\|I_m - \Lambda\right\|\left\|\hat{W}\right\|\left\|\vec{\sigma}'\left(\hat{V}^T \mu\right)\Delta V^T \mu\right\| + \|\Lambda\|\left(\|\Omega\| + \varepsilon_0^{\max}\right)\right)
\end{aligned} \tag{B.29}$$

Since the sigmoid $\vec{\sigma}$ and its derivative $\vec{\sigma}'$ are bounded functions of their arguments, an upper bound for the second order terms in the Taylor series expansion (B.19) can be computed.

$$\begin{aligned}
&\left\|O\left(\left\|\Delta V^T \mu\right\|^2\right)\right\| = \left\|\vec{\sigma}\left(\hat{V}^T \mu\right) - \vec{\sigma}\left(V^T \mu\right) - \vec{\sigma}'\left(\hat{V}^T \mu\right)\Delta V^T \mu\right\| \\
&\le 2\,\vec{\sigma}_{\max} + \left\|\vec{\sigma}'\left(\hat{V}^T \mu\right)\Delta V^T \mu\right\|
\end{aligned} \tag{B.30}$$

Using (B.30) an upper bound for Ω can be obtained.

$$\begin{aligned}\|\Omega\| &= \left\|\Delta W^T \vec{\sigma}'\left(\hat{V}^T\mu\right)V^T\mu + W^T O\left(\left\|\Delta V^T\mu\right\|^2\right)\right\| \\ &\le \|\Delta W\|\left\|\vec{\sigma}'\left(\hat{V}^T\mu\right)V^T\mu\right\| + \|W\|\left(2\bar{\sigma}_{\max} + \left\|\vec{\sigma}'\left(\hat{V}^T\mu\right)\Delta V^T\mu\right\|\right) \le \Omega_{\max}\end{aligned} \tag{B.31}$$

The uniform constant upper bound $\Omega_{\max}$ in (B.31) is due to the local properties of the sigmoidal function σ and the projection operator used in adaptive laws (B.28).

Similar arguments applied to (B.29) yield:

$$\begin{aligned}&\dot{V}\left(e, \Delta K_x, \Delta K_r, \Delta W, \Delta V\right) \\ &\le -\lambda_{\min}(Q)\|e\|^2 + 2\|e\|\|P B_1\|\left(\|I_m - \Lambda\|c_1 + \|\Lambda\|\left(\Omega_{\max} + \varepsilon_0^{\max}\right)\right)\end{aligned} \tag{B.32}$$

where positive constant c_1 is chosen such that $\left\|\hat{W}\right\|\left\|\vec{\sigma}'\left(\hat{V}^T\mu\right)\Delta V^T\mu\right\| \le c_1$. From (B.32) it immediately follows that the time derivative of the Lyapunov function is negative definite outside of the following compact set:

$$\left\{e \in R^n : \|e\| \le \frac{2\|P B_1\|\left(\|I_m - \Lambda\|c_1 + \|\Lambda\|\left(\Omega_{\max} + \varepsilon_0^{\max}\right)\right)}{\lambda_{\min}(Q)}\right\} \tag{B.33}$$

This, in turn, implies bounded tracking of the reference signal. In addition, since the projection operator bounds the SHL NN estimated weighs W, V and their increments $\Delta W, \Delta V$, all the signals in the corresponding closed-loop system remain bounded.

APPENDIX C
APPROXIMATE SOLUTION OF THE ALGEBRAIC LYAPUNOV EQUATION

The Lyapunov Algebraic Equation (LAE) is of the form:

$$0 = P A + A^T P + Q \tag{C.1}$$

where A is the (n-by-n) – *Hurwitz* matrix (all eigenvalues of A are in the left half plane), Q is the (n-by-n) – *symmetric positive definite* matrix, and P is the (n-by-n) – *symmetric positive-definte* solution matrix. The exact solution of (C.1) is:

$$P = \int_0^\infty \exp\left(A^T \tau\right) Q \exp\left(A \tau\right) d\tau \tag{C.2}$$

Let

$$\hat{P}(t) = \int_0^t \exp\left(A^T \tau\right) Q \exp\left(A \tau\right) d\tau \tag{C.3}$$

Then

$$\hat{P} A + A^T \hat{P} = \int_0^t \left\{ \exp\left(A^T \tau\right) Q \exp\left(A \tau\right) A + A^T \exp\left(A^T \tau\right) Q \exp\left(A \tau\right) \right\} d\tau$$

$$= \int_0^t \left[\frac{d\left\{ \exp\left(A^T \tau\right) Q \exp\left(A \ \tau\right) \right\}}{d\tau} \right] d\tau = \underbrace{\exp\left(A^T t\right) Q \exp\left(A t\right)}_{\dot{\hat{P}}} - Q = \dot{\hat{P}} - Q$$

The resulting equation

$$\dot{\hat{P}} = \hat{P} A + A^T \hat{P} + Q \tag{C.4}$$

is called the Lyapunov Differential (continuous) Equation (LDE). From (C.2) and (C.3) it follows that

$$\lim_{t \to \infty} \left\| \hat{P}(t) \right\| = P \tag{C.5}$$

In other words, the LDE solution $\hat{P}(t)$ approximates, (*i.e.*, asymptotically tends to), the corresponding LAE solution P, forward in time.

Conclusion: Numerical integration of LDE (C.4) yields an approximation to the true LAE solution.

Example: Using AB-2 integration method, we get

$$\begin{cases} \dot{\hat{P}}(t_k) = \hat{P}(t_k) A + A^T \hat{P}(t_k) + Q \\ \hat{P}(t_{k+1}) = \hat{P}(t_k) + \Delta t \left(1.5 \dot{\hat{P}}(t_k) - 0.5 \dot{\hat{P}}(t_{k-1}) \right), \quad k = 1, \ 2, \ \ldots \end{cases} \tag{C.6}$$